

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

POSITION ESTIMATION FROM RANGE ONLY MEASUREMENTS

by

Jason C. Alleyne

September 2000

Thesis Advisor:

Anthony J. Healey

Approved for public release; distribution is unlimited.

20010215 025

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2000	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: Position Estimation From Range Only Measurements			5. FUNDING NUMBERS N0001499AF00002	
6. AUTHOR(S) JASON C. ALLEYNE				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER N/A	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY REPORT NUMBER N/A	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) In order for a team of several Automated Underwater Vehicles (AUVs), such as the ARIES, to operate cooperatively, operators require a cost effective position estimation method. Range only measurement (ROM) position estimation provides this and a means for the AUVs to identify each other's position. Position estimation usually requires at least two range measurements from known points to solve for a vessel's position. However, under certain conditions, one range only measurement can provide a simpler solution. This thesis proves ROM as a viable means of target tracking and position estimation. Determining the accuracy and observability of ROM serve as the primary focus. The ROM model setup and execution are discussed with specific attention given to the details of the Extended Kalman Filter (EKF) and calculations required to determine the system's observability.				
14. SUBJECT TERMS: Autonomous Underwater Vehicles, Unmanned Underwater Vehicles, Robotics, Navigation			15. NUMBER OF PAGES 160	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited.

POSITION ESTIMATION FROM RANGE ONLY MEASUREMENTS

Jason C. Alleyne
Lieutenant, United States Navy
B.S., United States Naval Academy, 1993

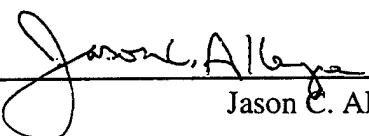
Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

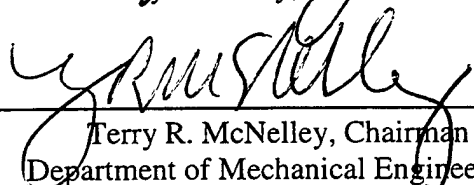
NAVAL POSTGRADUATE SCHOOL
September 2000

Author:


Jason C. Alleyne

Approved by:


Anthony J. Healey, Thesis Advisor


Terry R. McNelley, Chairman
Department of Mechanical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In order for a team of several Automated Underwater Vehicles (AUVs), such as the ARIES, to operate cooperatively, operators require a cost effective position estimation method. Range only measurement (ROM) position estimation provides this and a means for the AUVs to identify each other's position. Position estimation usually requires at least two range measurements from known points to solve for a vessel's position. However, under certain conditions, one range only measurement can provide a simpler solution. This thesis proves ROM as a viable means of target tracking and position estimation. Determining the accuracy and observability of ROM serve as the primary focus. The ROM model setup and execution are discussed with specific attention given to the details of the Extended Kalman Filter (EKF) and calculations required to determine the system's observability.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION	
A.	BACKGROUND.....	1
B.	SCOPE OF THIS WORK	3
II.	OVERVIEW OF ARIES	5
A.	NAVIGATION.....	5
B.	COMMUNICATIONS	6
C.	SERVER VEHICLE CONCEPT	6
III.	RANGE ONLY MEASUREMENT (ROM) THEORY.....	11
A.	INTRODUCTION.....	11
B.	PROBLEM FORMULATION	11
1.	The Dynamic Model.....	12
2.	Kalman Filtering.....	15
3.	Observability Determination	18
IV.	RESULTS OF SIMULATIONS	23
A.	INTRODUCTION.....	23
B.	CASE ONE MODEL	23
1.	Observability	24
2.	ROM Accuracy.....	24
C.	CASE TWO MODEL	25
1.	Observability	26
2.	ROM Accuracy.....	26
V.	CONCLUSIONS AND RECOMMENDATIONS.....	73
A.	CONCLUSIONS	73
B.	RECOMMENDATIONS	73
	APPENDIX A. <i>MATLAB</i> CODE ASSOCIATED WITH KNOWN INITIAL POSITION SIMULATION	75

APPENDIX B. <i>MATLAB</i> CODE ASSOCIATED WITH UNKNOWN INITIAL POSITION SIMULATION	81
APPENDIX C. <i>MATLAB</i> CODE ASSOCIATED WITH EXTENDED KALMAN FILTER MEASUREMENT DATA FUSION	87
LIST OF REFERENCES	89
INITIAL DISTRIBUTION LIST	91

LIST OF FIGURES

Figure 2.1	Hardware Components of the NPS ARIES	8
Figure 2.2	Server vehicle concept.....	9
Figure 3.1	AUV Paths with known "drone" intial position	20
Figure 3.2	AUV Paths with random, unknown "drone" intial position	21
Figure 4.1	Local Observability for all "master" AUV path radii.....	29
Figure 4.2	Total Observability for all "master" AUV path radii	30
Figure 4.3	East-west EKF position estimates vs. actual position for 100 meter "master" AUV path radius.....	31
Figure 4.4	North-south EKF position estimates vs. actual position for 100 meter "master" AUV path radius.....	32
Figure 4.5	East-west EKF position estimates vs. actual position for 50 meter "master" AUV path radius.....	33
Figure 4.6	North-south EKF position estimates vs. actual position for 50 meter "master" AUV path radius.....	34
Figure 4.7	East-west EKF position estimates vs. actual position for 30 meter "master" AUV path radius.....	35
Figure 4.8	North-south EKF position estimates vs. actual position for 30 meter "master" AUV path radius.....	36
Figure 4.9	East-west EKF position estimates vs. actual position for 17 meter "master" AUV path radius.....	37
Figure 4.10	North-south EKF position estimates vs. actual position for 17 meter "master" AUV path radius.....	38
Figure 4.11	AUV paths over 3000 seconds	39
Figure 4.12	Normalized differences between indicated and actual position for 100 meter "master" AUV path radius.....	40
Figure 4.13	Difference between actual and estimated position states for 100 meter "master" AUV path radius.....	41
Figure 4.14	AUV paths over 2000 seconds	42
Figure 4.15	Normalized differences between indicated and actual position for 50 meter "master" AUV path radius.....	43
Figure 4.16	Difference between actual and estimated position states with 50 meter "master" AUV path radius.....	44
Figure 4.17	AUV paths over 2000 seconds	45
Figure 4.18	Normalized differences between indicated and actual position for 30 meter "master" AUV path radius.....	46
Figure 4.19	Difference between actual and estimated position states with 30 meter "master" AUV path radius.....	47
Figure 4.20	AUV paths over 2000 seconds	48
Figure 4.21	Normalized differences between indicated and actual position for 17 meter "master" AUV path radius.....	49
Figure 4.22	Difference between actual and estimated position states with 17 meter "master" AUV path radius.....	50

Figure 4.23	Local Observability for all "master" AUV path radii with no drone initial condition data.....	51
Figure 4.24	Total Observability for all "master" AUV path radii with no drone initial condition data.....	52
Figure 4.25	East-west EKF position estimates vs. actual position for 100 meter "master" AUV path radius with no initial position data.....	53
Figure 4.26	North-south EKF position estimates vs. actual position for 100 meter "master" AUV path radius with no initial position data.....	54
Figure 4.27	East-west EKF position estimates vs. actual position for 50 meter "master" AUV path radius with no initial position data.....	55
Figure 4.28	North-south EKF position estimates vs. actual position for 50 meter "master" AUV path radius with no initial position data.....	56
Figure 4.29	East-west EKF position estimates vs. actual position for 30 meter "master" AUV path radius with no initial position data.....	57
Figure 4.30	North-south EKF position estimates vs. actual position for 30 meter "master" AUV path radius with no initial position data.....	58
Figure 4.31	East-west EKF position estimates vs. actual position for 17 meter "master" AUV path radius with no initial position data.....	59
Figure 4.32	North-south EKF position estimates vs. actual position for 17 meter "master" AUV path radius with no initial position data.....	60
Figure 4.33	AUV paths with no initial drone position data over 3000 seconds.....	61
Figure 4.34	Normalized differences between indicated and actual position for 100 meter "master" AUV path radius.....	62
Figure 4.35	Difference between actual and estimated position states for 100 meter "master" AUV path radius.....	63
Figure 4.36	AUV paths with no initial drone position data over 2000 seconds.....	64
Figure 4.37	Normalized differences between indicated and actual position for 50 meter "master" AUV path radius.....	65
Figure 4.38	Difference between actual and estimated position states for 50 meter "master" AUV path radius.....	66
Figure 4.39	AUV paths with no initial drone position data over 2000 seconds.....	67
Figure 4.40	Normalized differences between indicated and actual position for 30 meter "master" AUV path radius.....	68
Figure 4.41	Difference between actual and estimated position states for 30 meter "master" AUV path radius.....	69
Figure 4.42	AUV paths with no initial drone position data over 2000 seconds.....	70
Figure 4.43	Normalized differences between indicated and actual position for 17 meter "master" AUV path radius.....	71
Figure 4.44	Difference between actual and estimated position states for 17 meter "master" AUV path radius.....	72

ACKNOWLEDGMENTS

I would like to thank the staff members, faculty, and fellow students who supported me during my time here at the Naval Postgraduate School; especially, Professor Tony Healey for his guidance and support during the thesis process. Additionally, I would like to acknowledge David Marco, PhD, and Robert Dayap, for the invaluable insight they imparted on me.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

The number of Third World/non-super power countries exercising their armed forces is increasing as border disputes, religious difference, and etc. fuel aggression. Although a majority of these nations have no ill intentions towards the United States, they may pose a very dangerous threat to U.S. vessels. By using minefields, a nation can stifle the use of a waterway. During the Persian Gulf War, mine fields were laid in the Persian Gulf by Iraqi forces. As a result of the mine layings, sea-lanes were severely restricted until proper mine clearing of frequented water ways could be completed. To clear minefields the Navy currently uses specialized ships, aircraft, and personnel. In any of the above methods, there is a very high risk of human life loss in the event of an accident. As a result, alternate means of conducting mine clearing tasks have been explored including the use of marine mammals and Autonomous Underwater Vehicles (AUVs) /Unmanned Underwater Vehicles (UUVs). Marine mammal systems have the drawbacks of limited numbers, and extensive, expensive, training requirements. Additionally, their use would likely cause opposition from the public. Fortunately, AUVs are designed to be low maintenance, low cost, 'expendable' vehicles designed to carry out the mission in its program.

AUVs and UUVs are rapidly finding increased potential in military applications. From deep-water salvage to mine hunting/clearing operations, this new breed of vehicles provides a means for dangerous missions to be carried out without endangering personnel. More specifically, use of low cost AUVs in mine hunting/clearing operations

appears specifically attractive since there is no risk of human life loss or damaging expensive ships.

It is proposed that teams of AUVs working together may be used to search and identify large minefields. Such a team would consist of several "worker" vehicles and one "master" vehicle. The "worker" vehicles are expendable, low cost, AUVs designed to locate and possibly detonate enemy mines. The "master" vehicles are better equipped with more complex navigation, sonar, and communication suites. Pre-programmed "workers" would carry out their specific missions while the "master" vehicle would track and direct the drones to their specific waypoints. For a "master" vehicle to track several "workers", problems arise for a small, weight-conscious, AUV. Conventional position estimation in underwater applications usually consists of either active or passive towed array sonar. Unfortunately each application has drawbacks that preclude their use onboard the AUV. Active sonar would provide the most up to date and accurate positions of the worker vehicles. However, the transducer required to provide accurate positions of all the workers would be too large, require too much power, and be cost prohibitive for the size required in an AUV. Passive sonar is not a good choice for the AUV due to the time-late properties of its information. Additionally, vehicle mounted listening equipment that is of acceptable size, quality, and range is cost prohibitive, and use of a passive array requires handling equipment, which is not conducive to use in an AUV. Both passive and active tracking methods lack a means to explicitly discern mine-like objects from friendly AUVs.

In an effort to conserve space and weight, a means of Range only Measurement (ROM) position estimation is proposed. Song (1999) has demonstrated the observability

of ROM target tracking through use of simulations. Use of ROM tracking combines aspects of conventional active and passive tracking techniques. Like an active sensor, sound signals are emitted into the water using space and weight conscious acoustic modems transmit data through the water among vehicles in the team. Each signal contains “sent time”, course, and identification information. This information enables the “master” vehicle to estimate the distance between sender and receiver, and to identify each sender. Upon receiving the signal from the “master” vehicle, the drone vehicle transmits its signal to “master” vehicle. When the “master” vehicle receives this signal, the distance between the two vehicles can be estimated by calculating time of travel for the signal. To ensure maximum accuracy, the “master” vehicle would use water property values attained from ‘now’ data.

B. SCOPE OF THIS WORK

The overall problem of ROM position estimation is complex and diverse since it is usual that either range and bearing, or two range values are needed for position estimation. This study includes the development of the dynamic model and an Extended Kalman Filter (EKF) to demonstrate the accuracy and viability of ROM techniques for position estimation. The purpose of this thesis is twofold:

1. To prove the observability of ROM tracking techniques.
2. To illustrate the accuracy of ROM tracking by comparing it to dead reckoning, a common means of simple navigation.

Chapter II will provide the background of the Naval Postgraduate School (NPS) Acoustic Reconnaissance Interactive Exploratory Server (ARIES) AUV including navigation and sensor capabilities that are the basis of the “master” AUV’s capabilities.

Chapter III explains the theory and modeling of the simulation with detailed descriptions of the construction of the Extended Kalman Filter (EKF), and the means of determining the system's observability. Chapter IV summarizes the results of the simulations. Chapter V presents the conclusions gathered from the results, and offers recommendations for continued research on this topic.

II. OVERVIEW OF ARIES

A brief overview of the Naval Postgraduate School ARIES AUV shall provide some the working knowledge necessary to understand why certain assumptions were made, and to familiarize the reader with the AUV.

ARIES weighs 225 kilograms and measures about three m long, 0.4 m wide and 0.25 m high. The hull is constructed of one-quarter inch thick aluminum and forms the main pressure vessel that houses all electronics, computers, and batteries. External sensors are housed in the flooded fiberglass nose. Figure 2.1 shows the ARIES component layout. The vehicle has a top speed of three and one-half knots, which it may maintain for approximately four hours. The ARIES was primarily designed for shallow water operations and can operate safely down to 30 meters (Marco, Healey, 2000).

A. NAVIGATION

The sensor suite used for navigation includes a RD Instruments Navigator Doppler Velocity Log (DVL) that also contains a magnetic compass. This instrument measures the vehicle ground speed, altitude, and magnetic heading. Angular rates and accelerations are measured using a 3-axis Motion Pak Inertial Motion Unit (IMU). Data from the DVL and IMU are fused using an EKF to provide a high quality dead reckoning solution while submerged. While surfaced, ARIES utilizes Differential Global Positioning System (DGPS) to correct any navigational errors accumulated during the submerged phases of a mission.

The EKF in the ARIES' navigational suite may be tuned for optimal performance given a set of data. By fusing data input from the IMU / DVL / DGPS suite, biases, such

as yaw rate bias and compass bias, can be identified and compensated for during underwater travel. Although this compensation can not completely correct for environmental biases, a relatively short surface time, for example, 10 seconds allows the filter to re-estimate biases, correct position estimates and continue with improved accuracy (Marco, Healey, 2000).

For obstacle avoidance and target acquisition, the ARIES uses scanning or profiling sonar. The sonar heads can scan continuously through 360° of rotation or sweep through a defined angular sector (Marco, Healey, 2000).

B. COMMUNICATIONS

ARIES is equipped with two types of modems. Radio modems are used for high bandwidth command/control and system monitoring while the vehicle is surfaced. An acoustic modem is used for low bandwidth communications while the vehicle is submerged. Additionally, the acoustic modem enables ARIES to exchange pertinent data with other AUVs that have compatible components (Marco, Healey, 2000).

C. SERVER VEHICLE CONCEPT

This thesis models the “master” vehicle the server of a two vehicle network. Proposals for using the NPS ARIES as a network server vehicle for multi-vehicle cooperative operations have surfaced due to the foreseen benefits. Use of the server vehicle as a data relay increases the range of communications of the underwater components of the network. Figure 2.2 describes the concept where in position 1, the ARIES communicates through its acoustic modem with multiple worker vehicles that are

engaged in a search pattern. Position 2 shows the ARIES on the surface using a radio modem to report mission status of the worker vehicles (possibly vehicle positions, image snippets of targets, and hydrographic data) to the command ship. While surfaced the server vehicle can receive tactical decision re-tasking commands. Once the new orders are received, the vehicle will submerge and transmit, using its acoustic modem, new tasks to each worker vehicle. Using a server vehicle eliminates the complexity of deploying fixed buoys. Also, a vehicle of this type can achieve close proximity or rendezvous with the worker vehicles allowing for higher acoustic bandwidth data transfer (Marco, Healey, 2000).

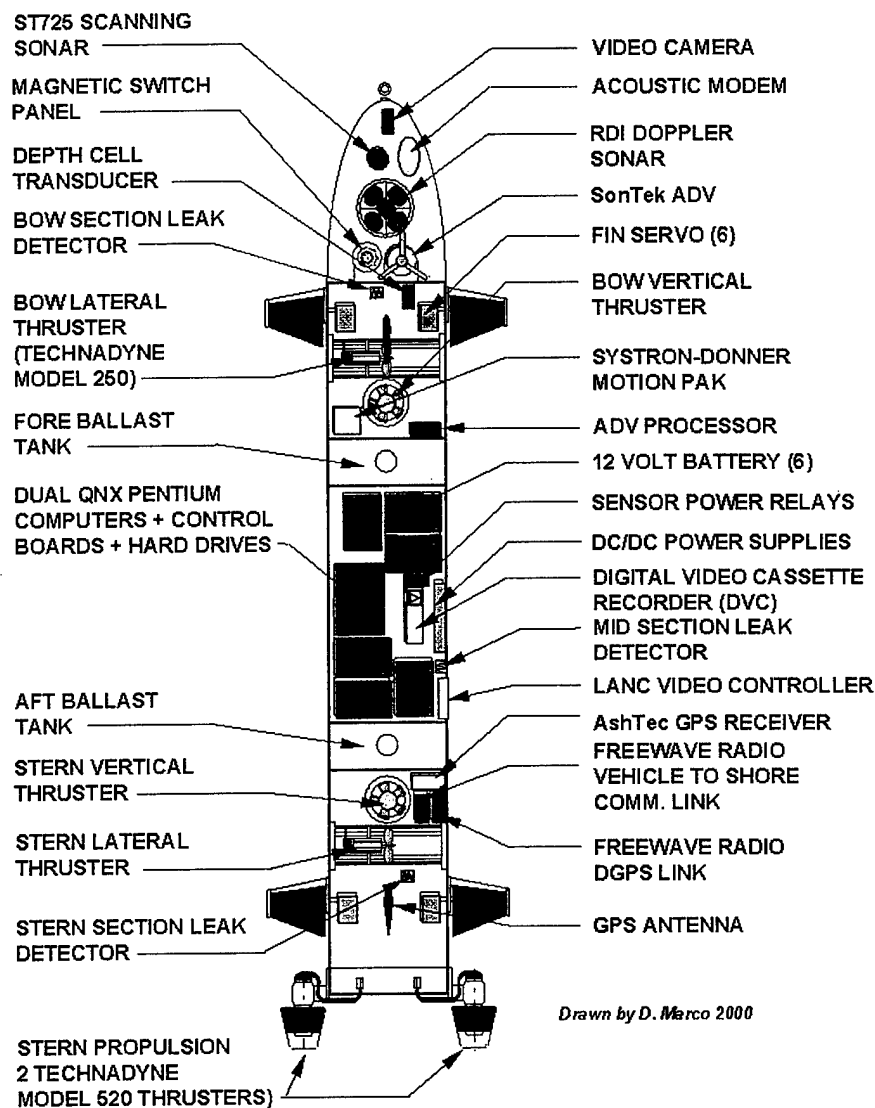


Figure 2.1 - Hardware Components of the NPS ARIES (Marco, Healey, 2000)

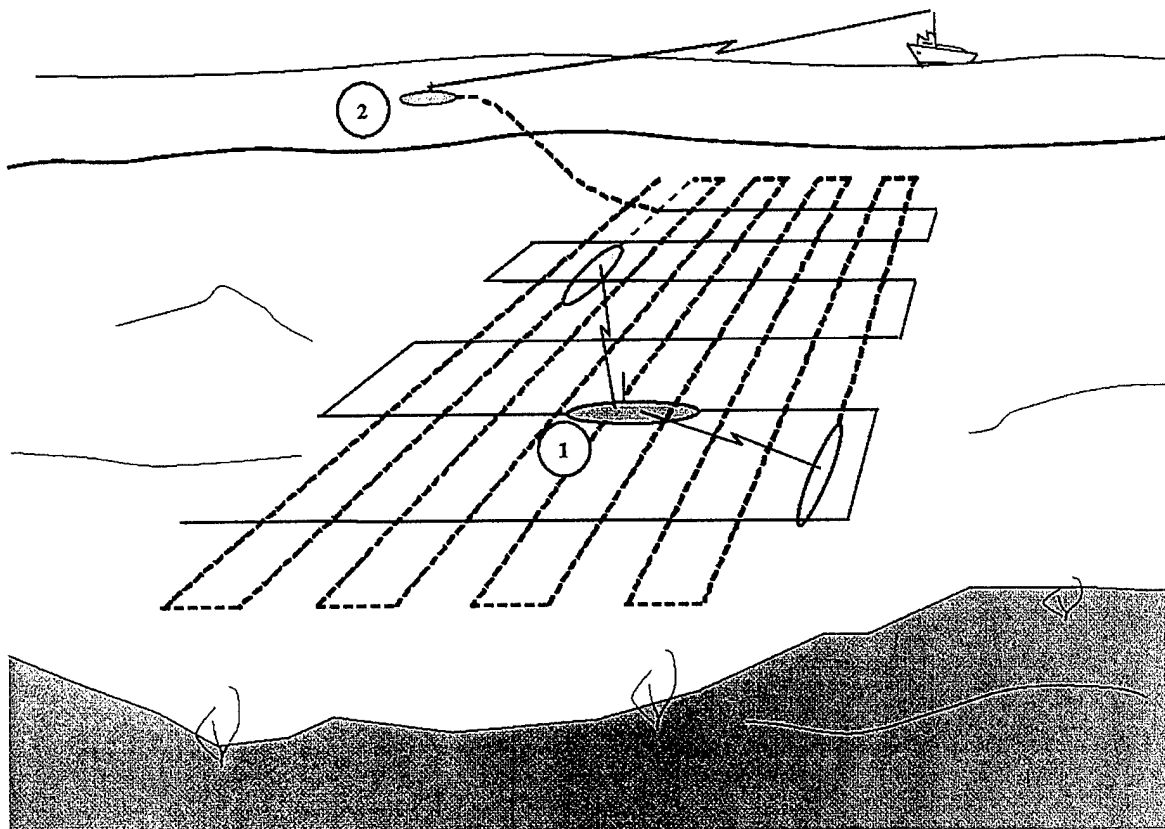


Figure 2.2 - Sever vehicle concept. 1. Low bandwidth submerged data transfer between underwater vehicles. 2. High-speed data relay to command ship (Marco, Healey, 2000).

THIS PAGE INTENTIONALLY LEFT BLANK

III. RANGE ONLY MEASUREMENT (ROM) THEORY

A. INTRODUCTION

Only in the past few years has serious research in Range Only Measurement (ROM) target tracking been conducted. Seen as a means to reduce the cost of active target tracking, methods of ROM target tracking are increasingly being investigated. ROM target tracking previously lacked interest of researchers due to "the lack of firm analytical basis for system observability in addition to the lack of proper filter structures in the field of target tracking with ROM from one observer." (Song, 1999)

Previous study also found that, "global system observability may not exist for the ROM target tracking; however, target state variables such as position, velocity, and acceleration can be estimated with a proper initial state estimate due to local system observability." (Song, 1999) If available, an initial state estimate can be obtained from previous target data, an update from an external source, i.e. a surface ship, or through the use of the "master" vehicle's active sonar.

This study furthers the above by showing that accurate position estimates can be obtained without an accurate initial position, and that global observability for ROM target tracking is possible under certain conditions.

B. PROBLEM FORMULATION

To properly investigate the questions concerning ROM's accuracy and observability an appropriate model had to be constructed. Since this tracking technique

III. RANGE ONLY MEASUREMENT (ROM) THEORY

A. INTRODUCTION

Only in the past few years has serious research in Range Only Measurement (ROM) target tracking been conducted. Seen as a means to reduce the cost of active target tracking, methods of ROM target tracking are increasingly being investigated. ROM target tracking previously lacked interest of researchers due to "the lack of firm analytical basis for system observability in addition to the lack of proper filter structures in the field of target tracking with ROM from one observer." (Song, 1999)

Previous study also found that, "global system observability may not exist for the ROM target tracking; however, target state variables such as position, velocity, and acceleration can be estimated with a proper initial state estimate due to local system observability." (Song, 1999) If available, an initial state estimate can be obtained from previous target data, an update from an external source, i.e. a surface ship, or through the use of the "master" vehicle's active sonar.

This study furthers the above by showing that accurate position estimates can be obtained without an accurate initial position, and that global observability for ROM target tracking is possible under certain conditions.

B. PROBLEM FORMULATION

To properly investigate the questions concerning ROM's accuracy and observability an appropriate model had to be constructed. Since this tracking technique

may be used on an AUV, the NPS ARIES AUV provided the basis for model vehicles' capabilities and constraints.

Modeling of the ROM system was done in computer based simulations using *MATLAB* due to its strong processing and presentation capabilities. Two simulations were developed; each models the same problem with a few changed parameters. Development of the simulation models was split into two sections:

1. The dynamic model
2. The Extended Kalman Filter (EKF)

The development and theory behind constructing the above sections will be explained in detail.

1. The Dynamic Model

The simulations model a system consisting of two AUVs, a "master" and a "drone" vehicle operating in a zero current environment. The depths of the two vehicles were disregarded as this is assumed to be only a two dimensional problem. The "master" AUV navigation suite matches that of the ARIES described in the previous chapter, and is assumed to have accurate position, velocity, acceleration, and course data at all times. On the contrary, the "drone" vehicle possesses a rudimentary navigation suite consisting of an off the shelf compass and a bottom-mounted Doppler Sonar Velocity Log. Combining its compass heading, speed over ground, and an initial estimated position, the drone uses dead reckoning to estimate and update its position.

For both simulations, almost ideal surroundings were used to model the environment. As stated above, both simulations were modeled without current or other

external forces, i.e., wind, waves, etc. To introduce an error between the actual and the dead reckoning courses, the compass was assumed to have a constant bias of five degrees east. The bias may seem artificially large, but testing with the AIRES has proven that this value represents actual bias errors experienced using comparable compasses. Other environmental factors that would affect the vehicles' motion were made negligible in order to focus on the details more critical to this thesis.

To provide the best conditions for ROM observability, careful selection of vehicle paths was necessary especially that of the "master" vehicle. In previous study, Dr. Taek Song found that:

...if the tracker-target relative motion results in a constant bearing trajectory, or if the tracker is moving with a constant velocity or a constant acceleration, the system is not observable. This implies that the tracker motion should contain nonzero jerk to track a target with constant acceleration. (Song, 1998)

Based on this observation, it was concluded that a circular path of some sort at maximum cruising speed would provide the best results by satisfying Dr. Song's criteria in tracking linear paths. It was decided to make the "master"/tracker vehicle's path a circle. In doing this, the "master" AUV's bearing, opening, and closing rates relative to the "drone" AUV could be altered by changing its path radius. In all cases, the "master" AUV's speed of one and one-half meters per second remained constant.

A more straight forward task, the "drone"/target vehicle path selection represents an AUV conducting a portion of its minesweeping search pattern at a constant speed of two-tenths meters per second. For the Case One simulation, the "drone's" dead reckoning and actual positions start in the center of the "master" vehicles circular path. The desired and dead reckoning course is due north, course 000, but due to the compass

error the actual path taken by the “drone” is five degrees east of course 005. Figure 3.1 illustrates the three paths described above. The Case Two simulation assumes the actual initial position of the “drone” is unknown, but located near the circular path of the “master” vehicle. This was done using *MATLAB*’s “randn” function which produces random entries chosen from a normal distribution with a zero mean and a variance of one. As in the Case One simulation, the actual drone model moves from its initial position on a course of 005 due to its compass bias. The “drone’s” dead reckoning position starts in the center of the “master” vehicle’s path and follows the desired due north course. Figure 3.2 shows the AUV paths in a typical Case Two simulation.

For each simulation, a five element state vector in the Cartesian coordinates consisting of two “master” AUV position states, $[X, Y]$, two “drone” AUV’s position states, $[x_I, y_I]$, and the “drone” AUV’s heading state, $[\psi]$, represented the system. The continuous dynamic system is expressed as

$$\dot{x} = Ax + Bu$$

where $x = (X, Y, x_I, y_I, \psi)^T$,

$$A = \begin{bmatrix} 0 & \omega & 0 & 0 & 0 \\ -\omega & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.2 \\ 0 \end{bmatrix}$$

and

$$\omega = \frac{Rs}{u_m}$$

in which Rs equals the radius in meters of the “master” AUV’s continuous circular path at speed u_m . It was assumed that the tracking system dynamics are represented perfectly so that the process noise is not included in this analysis, (Song, 1999).

The next step in the problem formulation was converting the continuous dynamic system into a discrete dynamic system so it would be compatible with the EKF. *MATLAB*'s continuous to discrete function, "c2d", eased the process considerably. The basics of the "c2d" algorithms are taken from the *Control Toolbox* and described here for background on the process.

To convert a state space linear time invariant (LTI) system expressed as

$$\dot{x} = Ax + Bu$$

into a discrete time state space system represented as

$$x_{k+1} = \Phi x_k + \Gamma u_k$$

$$u_k = 1$$

MATLAB calculates the matrix exponential phi, $\Phi(dt)$, as

$$\Phi = e^{A \cdot dt}$$

dt = the sample time increment

and gamma, Γ , as a value that maps inputs to system response such that

$$\Gamma = (\Phi - I) A^{-1} B.$$

With the values of phi and gamma, recursive loops in the simulation program calculate the discrete vehicle states for each increment of the specified simulation time length, t .

2. Kalman Filtering

The Kalman filter is a set of mathematical equations that provides an efficient recursive solution of the least-squares method. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown (Bishop and Welch,

1995). The algorithm computes the best estimates of system variables arising from sensor based data and the system model. Data from measurements along with the measurement model are used in a system model to provide the least squares fit estimate of system state, based on those measurements (Healey, An, Marco 1998). The Kalman filter also assumes there is a zero mean error associated with the measurement data and estimation process. Based on *a priori* information, the modeler has the ability to adjust these error values to increase simulation accuracy.

Due to the nonlinear nature of equations used in the simulations, an EKF was used to fuse the available data and provide position estimates. The system is a continuous time model of vehicle motion represented by

$$\begin{aligned}\dot{x}(t) &= f(s(t)) + q(t); \\ y(t) &= h(x(t)) + v(t);\end{aligned}$$

where $x(t) \in \mathbb{R}^{5 \times 1}$ is the model state, f and h are continuous functions differentiable by $x(t)$, and q and v are zero mean white noise excitations for the system and measurement models respectively (Healey, An, Marco, 1998). For the “master” AUV, the states are globally referenced longitude and latitude in meters, X and Y , the “drone” AUV states are also globally referenced, actual, longitude and latitude in meters, x_l and y_l , and the heading angle referenced to North, ψ (Stinespring, 2000). The filter state vector, x , is made up of the above states, and the system is of order five.

$$x = [X, Y, x_l, y_l, \psi]^T;$$

The state model is related through the following set of functions representing dynamic relationships between states with assumptions embodying maneuvering models:

$$\begin{aligned}
\dot{X} &= \omega Y \\
\dot{Y} &= -\omega X \\
\dot{x}_1 &= 0; \\
\dot{y}_1 &= 0; \\
\dot{\psi} &= 0;
\end{aligned}$$

To calculate the above state functions and solve the equations, the EKF takes the outputs from some of the “master” AUV’s sensors measurements. The particular sensors that provide measurement data at that time are related to the filter state with the C matrix (Healey, An, Marco, 1998). This matrix is also used to convert estimated state vectors into the output matrix y . This matrix is in the form:

$$y^T = [Range, X, Y, x_1, y_1, \psi];$$

The equations, which are related in the C matrix and determine the output based on what the system is receiving from the sensors are as follows (Stinespring, 2000):

$$\begin{aligned}
y_1 &= \sqrt{(X_k - x_{1k})^2 + (Y_k - y_{1k})^2} = Range \\
y_2 &= X; \\
y_3 &= Y; \\
y_4 &= x_1; \\
y_5 &= y_1; \\
y_6 &= \psi;
\end{aligned}$$

The output y_1 is range between the “master” and “drone” vehicles at the time increment k , $y_{2,3}$ are the “master” vehicle’s position components as recorded from DGPS, $y_{4,5}$ are the “drone” position components as reported by their onboard position estimators, and y_6 is the “drone” AUV’s heading as reported from its compass. The output is then used to refine the drone position based on the sensor outputs. It was assumed that ranges calculated by the acoustic modem signal travel times would be accurate. Ranges used by the EKF in the simulations are exact and utilize data that would not be available to the filter in real world situations.

The C matrix composition is the only difference in the EKF between the Case One and Case Two simulations. In the Case One simulation, the C matrix's construction relates data from all sensors channels to EKF. This simulation provides more data than the EKF needs to provide an accurate position estimate, but it served as excellent program to tune the EKF. In the Case Two simulation, the C matrix only allows channels one through three, and six to be used by the EKF. The remaining channels were made unusable to the EKF by zeroing the applicable coefficients in the C matrix.

As stated before, the simulations represent near ideal systems. The system and measurement noise matrices of Q and R , respectively, were kept relatively small and can be considered to have little to no effect on the results for both simulations, except to determine the response time of the EKF.

3. Observability Determination

EKF solution observability is probably the most important parameter in whether the data from the EKF is valid. Observability of the filter is essential in order to guarantee stable unbiased estimation errors of the state. A nonlinear filter such as the one used in the simulations, can be verified for observability locally through linearization of the f and h functions (Healey, An, Marco, 1998). For a linearized model,

$$\begin{aligned}\dot{x}(t) &= Ax(t) + q; \\ y(t) &= Cx(t) + v;\end{aligned}$$

$$\text{where } A = \frac{\partial f}{\partial x} \text{ and } C = \frac{\partial h}{\partial x};$$

the system is locally observable if the following observability matrix, O , has full rank of five.

$$O = [C', C'A', C'A'^2, C'A'^3, C'A'^4, C'A'^5]$$

To calculate the total system observability, the integral of the observability grammian over the entire simulation time period, t , must be of full rank.

$$rank(\int_0^T (\Phi^T(\tau,0)C^T(\tau)C(\tau)\Phi(\tau,0))d\tau) = n;$$

$$\text{where } n = \text{length}(x);$$

Observability matrices of full rank in each simulation provide an excellent basis for establishing ROM's observability.

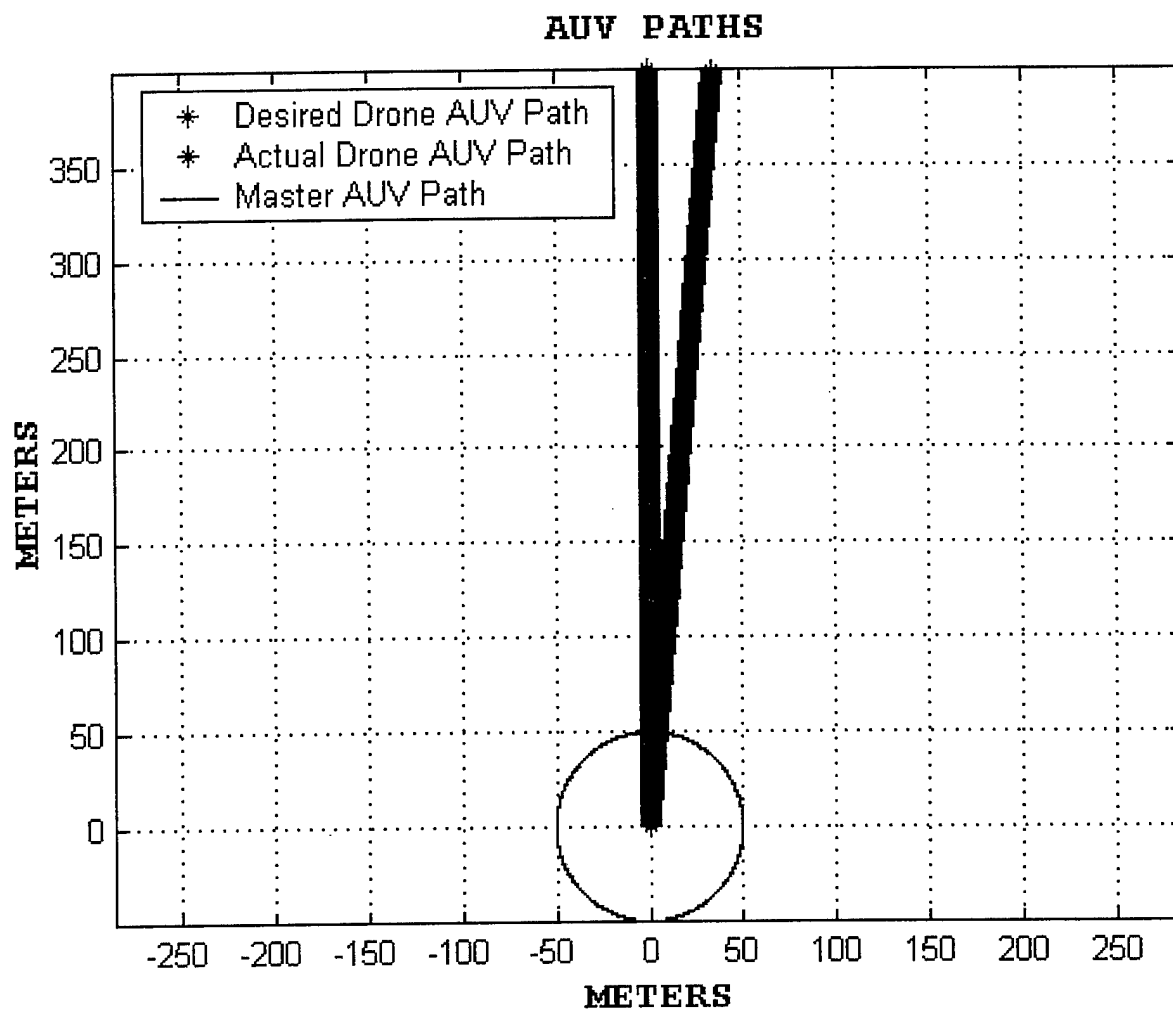


Figure 3.1 – AUV paths with known “drone” initial position

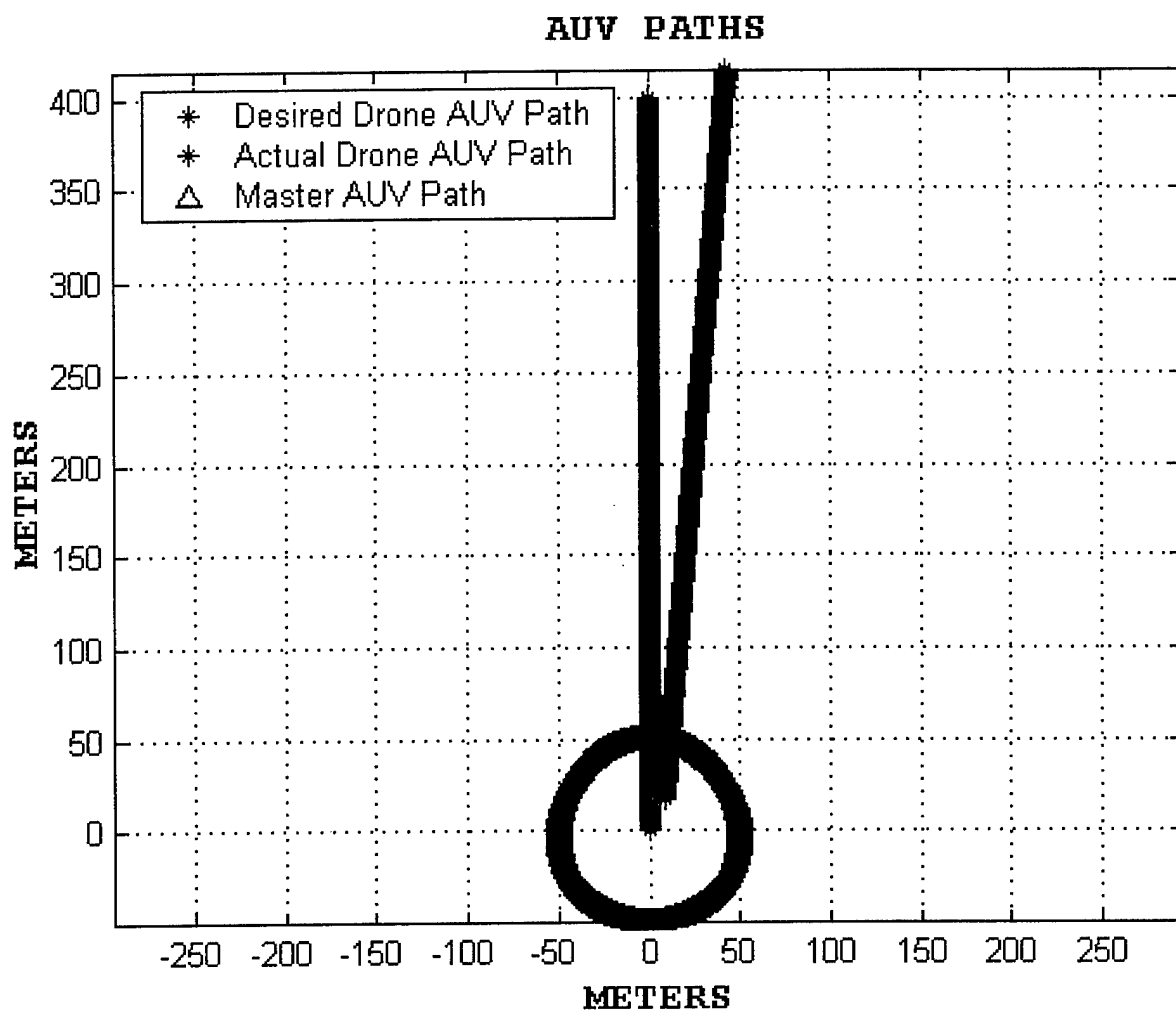


Figure 3.2 - AUV paths with random, unknown “drone” initial position

THIS PAGE INTENTIONALLY LEFT BLANK

IV. RESULTS OF SIMULATIONS

A. INTRODUCTION

During development and data collection, several server vehicle speeds and circular path radii were used. As a means to limit the amount of data collected, vehicle speeds were kept constant and turning capabilities were based on those of the AIRES AUV.

Two simulation scenarios were modeled to test the ROM EKF. In the Case One scenario, the starting position of drone vehicle was constant and accurately passed to the EKF. Additionally, by modifying the C matrix, described in the previous chapter, the EKF used all the channels of measured sensor data to more accurately calculate the “drone” AUV’s position. In the Case Two scenario, it was intended to model a system in which the starting position of the drone AUV was unknown, and its position data was not available to the EKF.

To further characterize the accuracy of the EKF, the simulations were run with “master” AUV path radii of 17, 30, 50, and 100 hundred meters. The speed of the “master” vehicle was kept constant at one and one-half meters per second, which is the AIRES’ maximum speed.

B. CASE ONE MODEL

In this scenario, the “drone” AUV’s actual and dead reckoning position started at the center “master” AUV’s circular path. As the simulation progresses, the desired and actual paths of the “drone” AUV diverge due to compass bias as shown in Figure 3.1.

The C matrix composition allows the actual position and heading states of the “drone” vehicle to pass to the EKF.

1. Observability

In all Case One simulations, the system exhibited complete observability. The system’s local observability matrix and observability grammian were always of full rank. This was to be expected since all of the filter states were updated by sensor data at every time increment. Figures 4.1 and 4.2 depict the local and total system observability, respectfully. These figures are applicable to all “master” AUV path radii conducted in this scenario.

2. ROM Accuracy

Configured to relate data from all measurement channels, the EKF made accurate position approximations, but a noticeable oscillatory error in the east-west/cross track accuracy plagued runs with larger “master” AUV paths. Further testing determined that the error was related to the speed in which the “master” AUV completed its circular path or in other words its cyclic speed. Since the “master” AUV’s speed was kept fixed, modifying its path radius was the only means to effect cyclic speed. Decreasing the radius of “master” AUV’s path increased its cyclic rate. Accuracy of the EKF improved dramatically as the cyclic rates of the “master” AUV were increased. The difference in position between the actual and estimated “drone” AUV position drops considerably as illustrated in the following figures. Figure 4.3 compares the actual east-west position to the EKF estimated position for a “master” AUV path radius of 100 meters. Figure 4.4

compares the actual north-south position to the EKF estimated position for a “master” AUV path radius of 100 meters. Figures 4.5 through 4.10 show the same information as figures 4.4 and 4.5, for the three other “master” AUV paths.

In comparing ROM position estimation to dead reckoning, ROM demonstrated itself as a superior. Unlike dead reckoning, which may produce unbounded position errors, ROM position estimation solutions converge if given sufficient time. For short periods of time, dead reckoning provided positions that rival the accuracy of ROM position estimation. However, for periods longer than approximately 1000 seconds, ROM position estimates mirror the “drone” AUV’s actual position much more closely than dead reckoning.

Figure 4.11 shows the paths of the “master” AUV, the “drone” AUV’s desired and actual paths, and the ROM position estimate path. This figure clearly depicts ROM’s accuracy over dead reckoning. To further show ROM’s accuracy over dead reckoning, Figure 4.12 illustrates the normalized differences between the dead reckoning and actual positions and the ROM estimated and actual positions. Figure 4.13 shows the error of ROM position estimation states is bounded, and over time decreases as it converges. Figures 4.14 through 4.22 illustrate the information displayed in Figures 4.11 through 4.13 for the other “master” AUV path radii. These figures clearly show that as the “master” AUV’s cyclic rate increases, the accuracy of position estimation also increases.

C. CASE TWO MODEL

It was intended for this portion of the simulation to model a scenario in which the “master” AUV did not have an accurate “drone” AUV initial position. To further

simulate real world conditions, the only data made available to the EKF was the “drone” AUV range and heading data and the “master” AUV’s position.

1. Observability

In all no dead-reckoning simulations, the rank of the local observability matrix at any point in time was four. Unlike the previous scenario, “drone” vehicle position was not one of the sensor measurements used to update the EKF. Therefore its states became locally unobservable. Using the observability grammian to calculate observability over the entire simulation time span, the number of observable states increased to five. Initially, the drone’s ability to send its heading seems trivial. However, without this data it would not be possible for the observability grammian to reach full rank. Figures 4.23 and 4.24 show the system’s local observability and total observability, respectively. These figures are applicable to all “master” AUV path radii conducted in this scenario.

2. ROM Accuracy

Unlike the simulations with known initial “drone” positions, the EKF did not produce accurate estimates in the early stages of the simulations. It generally took one “master” AUV cycle before gaining acceptable position data. Initially, the EKF appears slow to adjust to the “drone’s” actual position due to their proximity and minimal change in range, but once the range and range rate increase the EKF quickly “zeroes in” on the drone AUV’s actual position. As demonstrated in the previous model, when the cyclic rate of the “master” AUV increased, the accuracy of the EKF also increased. Additionally, increasing the “master” AUV’s cyclic rate allowed a faster acquisition of

the “drone” by the EKF. They become smaller and recover more quickly as cyclic rate increases. This is illustrated by characterizing the initial spikes in the EKF’s position estimation prior to obtaining an accurate solution in the following figures. Figure 4.25 compares the actual east-west position to the EKF estimated position for a “master” AUV path radius of 100 meters. Figure 4.26 compares the actual north-south position to the EKF estimated position for a “master” AUV path radius of 100 meters. Figures 4.27 through 4.32 show the same information as figures 4.25 and 4.26, for the three other “master” AUV paths. The above figures show that as the “master” vehicle’s path radius decreases, the initial position error displayed by the spikes lessens and accurate position estimates occur quicker.

Again, ROM target tracking proved to be a more accurate means of position estimation than dead reckoning for longer periods of time. For shorter periods of time, dead reckoning provided more accurate position estimations. Unlike the previous model, there was not a general time under which dead reckoning was more accurate. Since the EKF took approximately one cycle to build accurate position solutions, the time in which the dead reckoning solution was more accurate varied.

Once again, decreasing the “master” AUV path radius increased the accuracy of the EKF position estimations. This effect is shown clearly in the following figures. Due to the random initial position generated by the program, figures representing any single “master” AUV path radius may not be taken from the same simulation. Figure 4.33 shows the paths of the “master” AUV, and the “drone” AUV’s dead reckoning, estimated, and actual paths. Figure 4.34 shows the normalized difference between dead reckoning and actual position and the normalized difference between ROM and actual

positions. Figure 4.35 shows the error between the actual and estimated position states of the “drone” AUV. Figures 4.36 through 4.44 illustrate the information displayed in Figures 4.33 through 4.35 for the other “master” AUV path radii.

LOCAL SYSTEM OBSERVABILITY

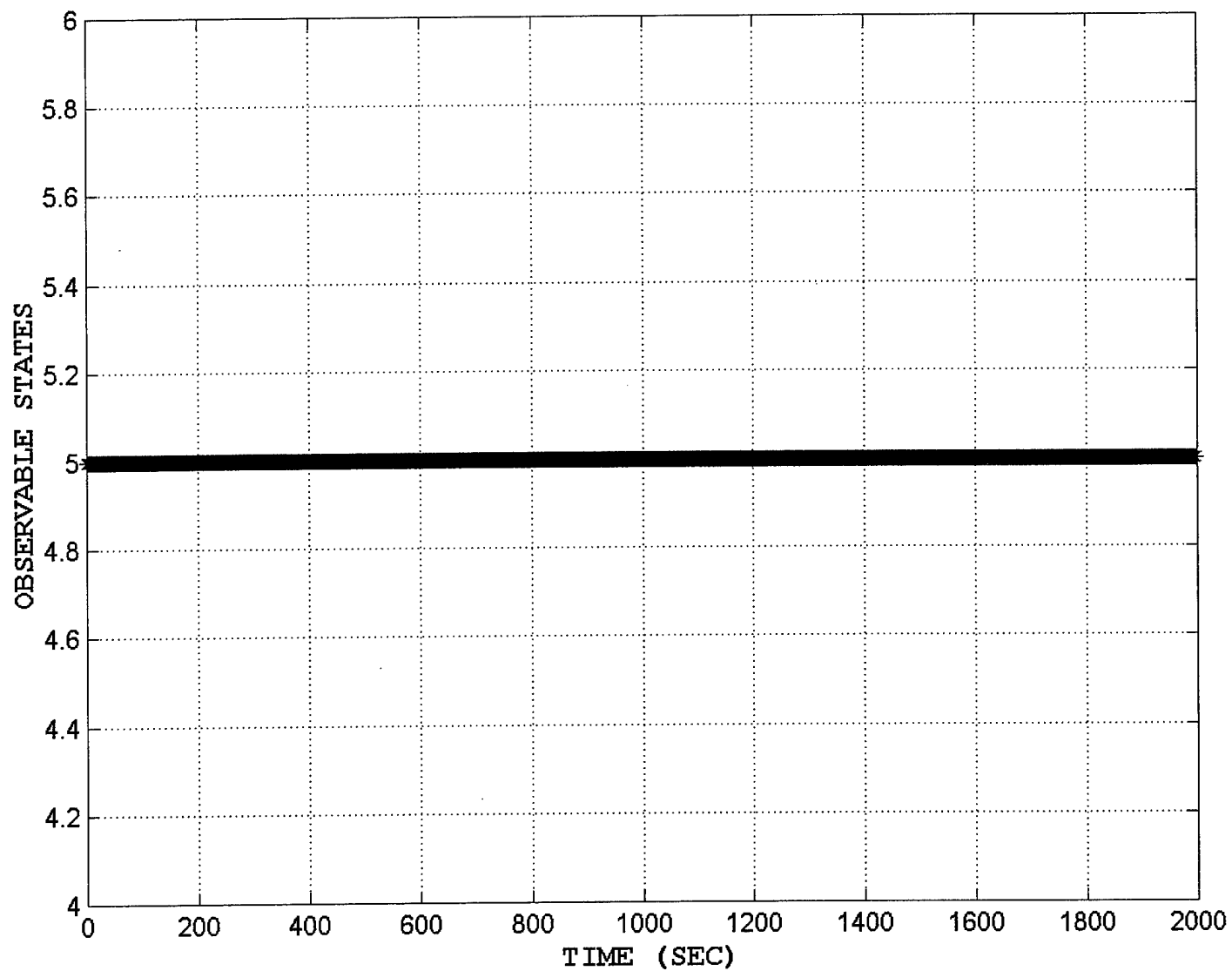


Figure 4.1 – Local Observability for all “master” AUV path radii

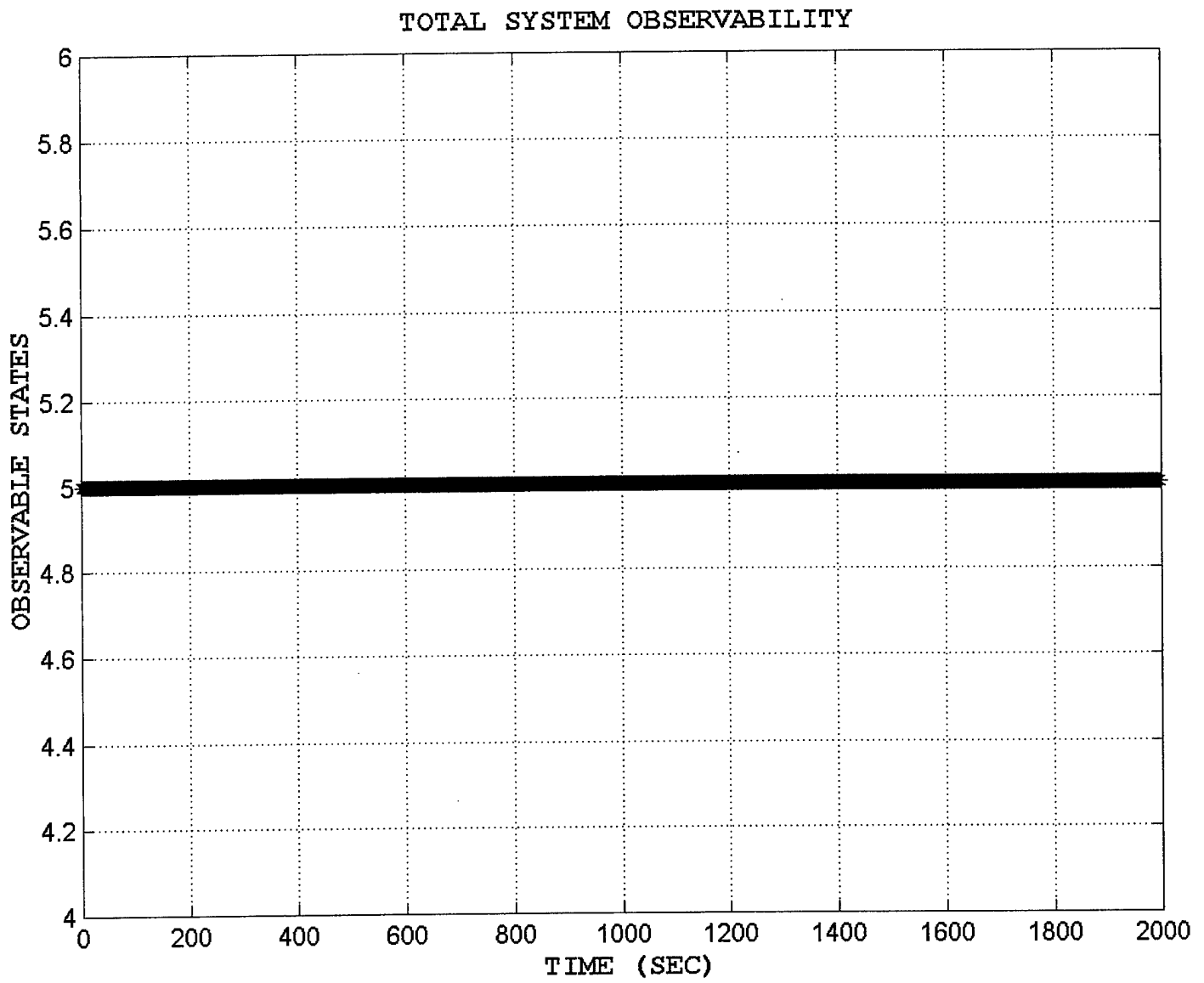


Figure 4.2 – Total observability for all “master” AUV path radii

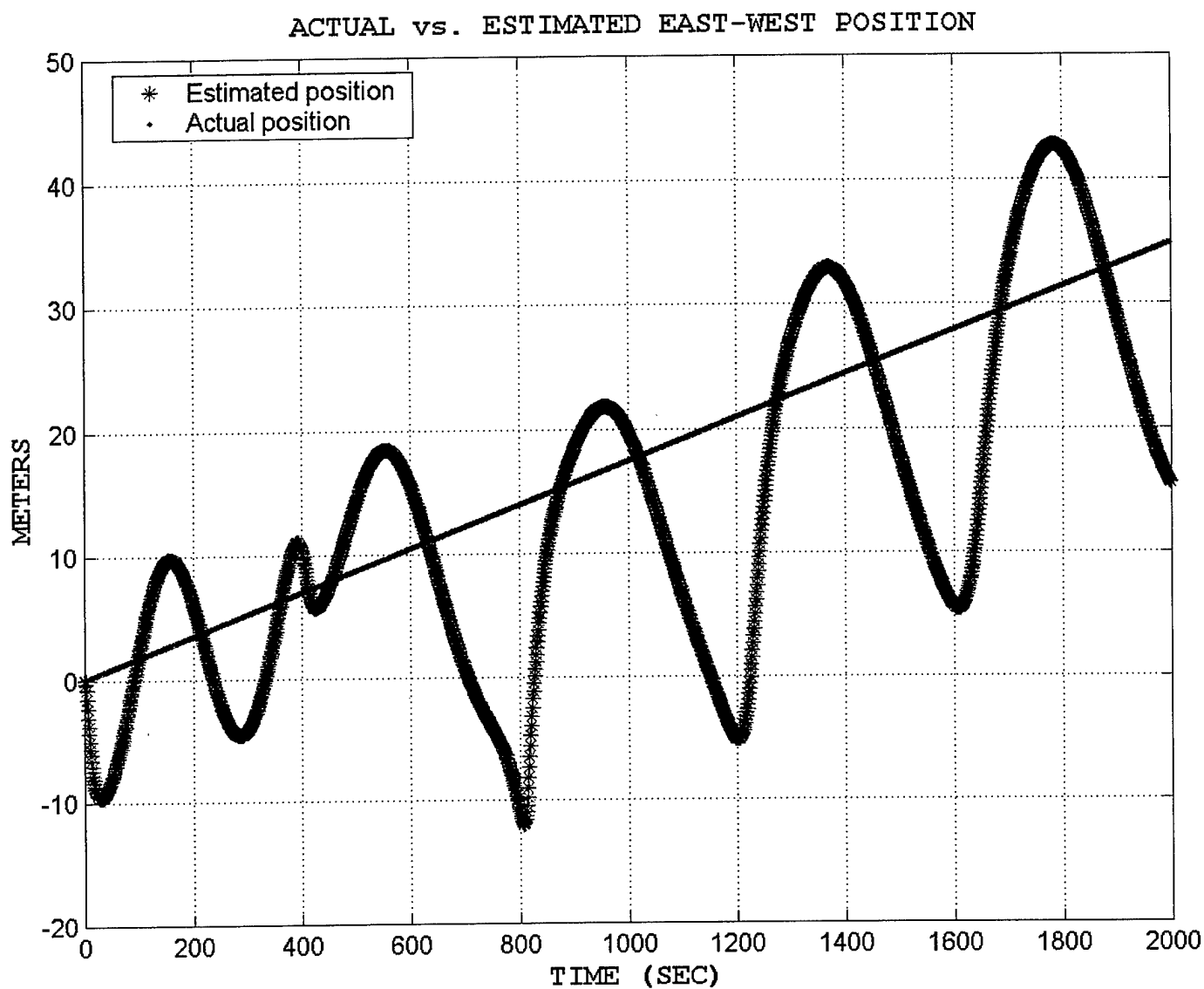


Figure 4.3 – East-west EKF position estimates vs. actual position for 100 meter “master” AUV path radius

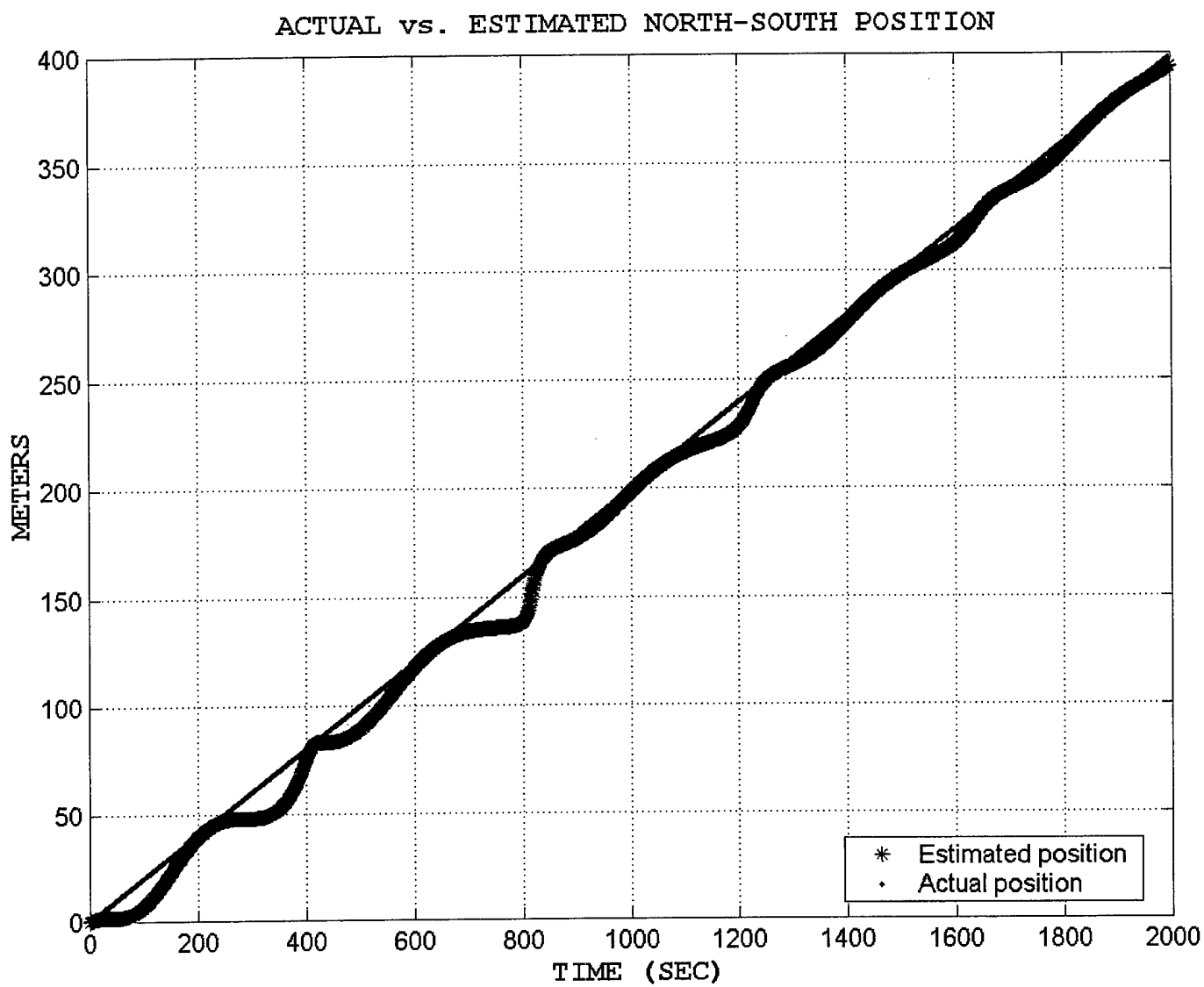


Figure 4.4 – North-south EKF position estimates vs. actual position for 100 meter “master” AUV path radius

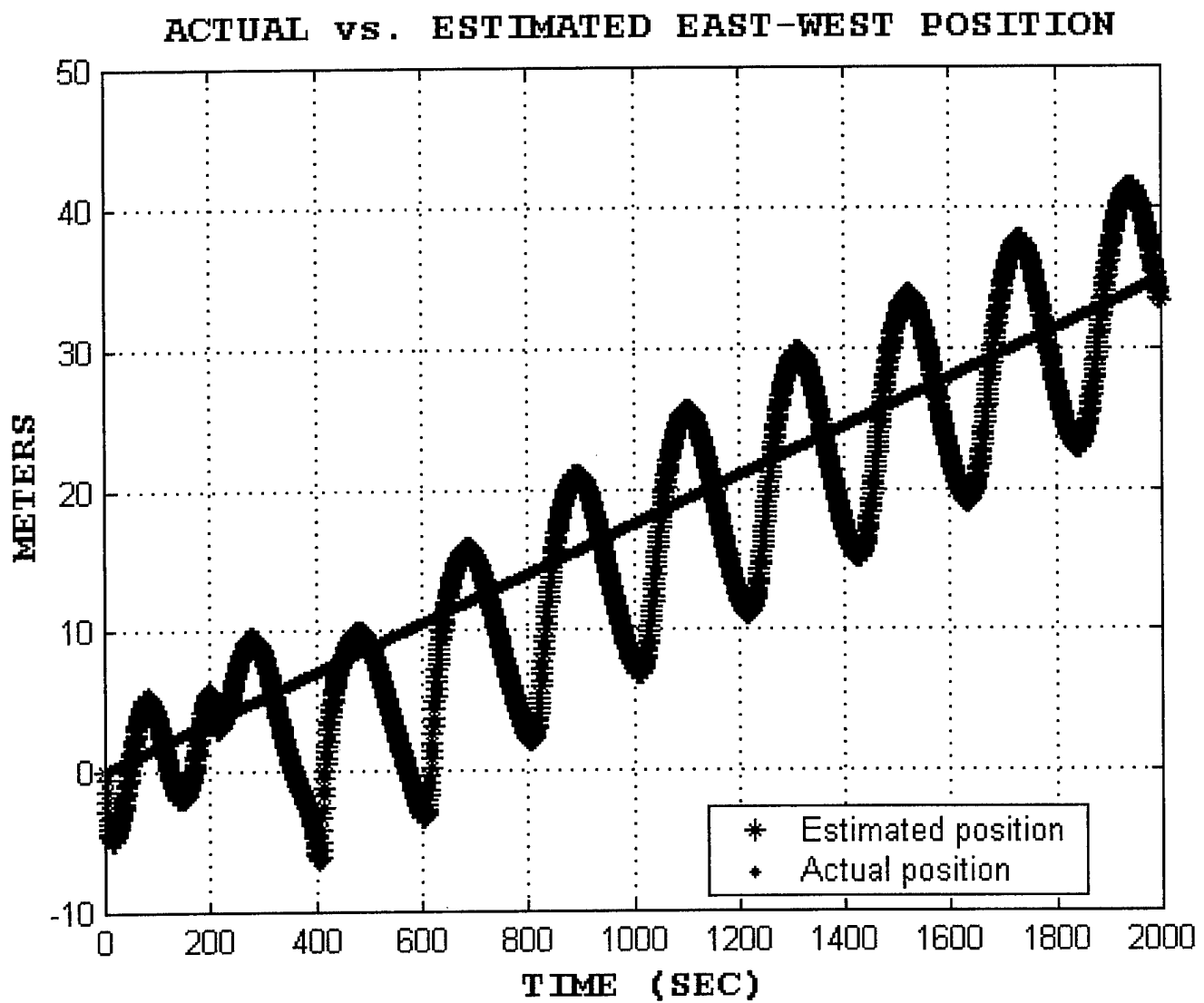


Figure 4.5 – East-west EKF position estimates vs. actual position for 50 meter
"master" AUV path radius

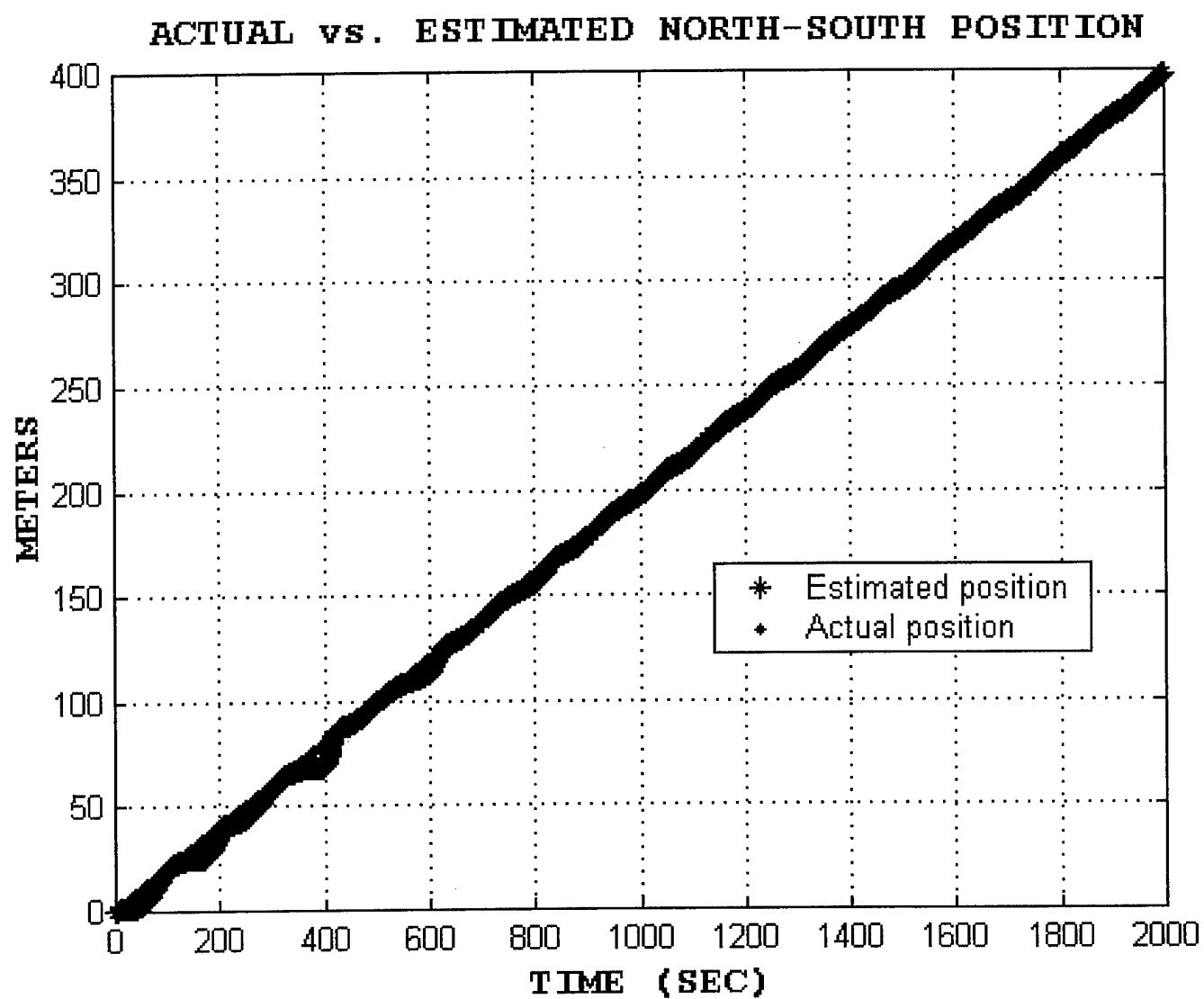


Figure 4.6 – North-south EKF position estimates vs. actual position for 50 meter “master” AUV path radius

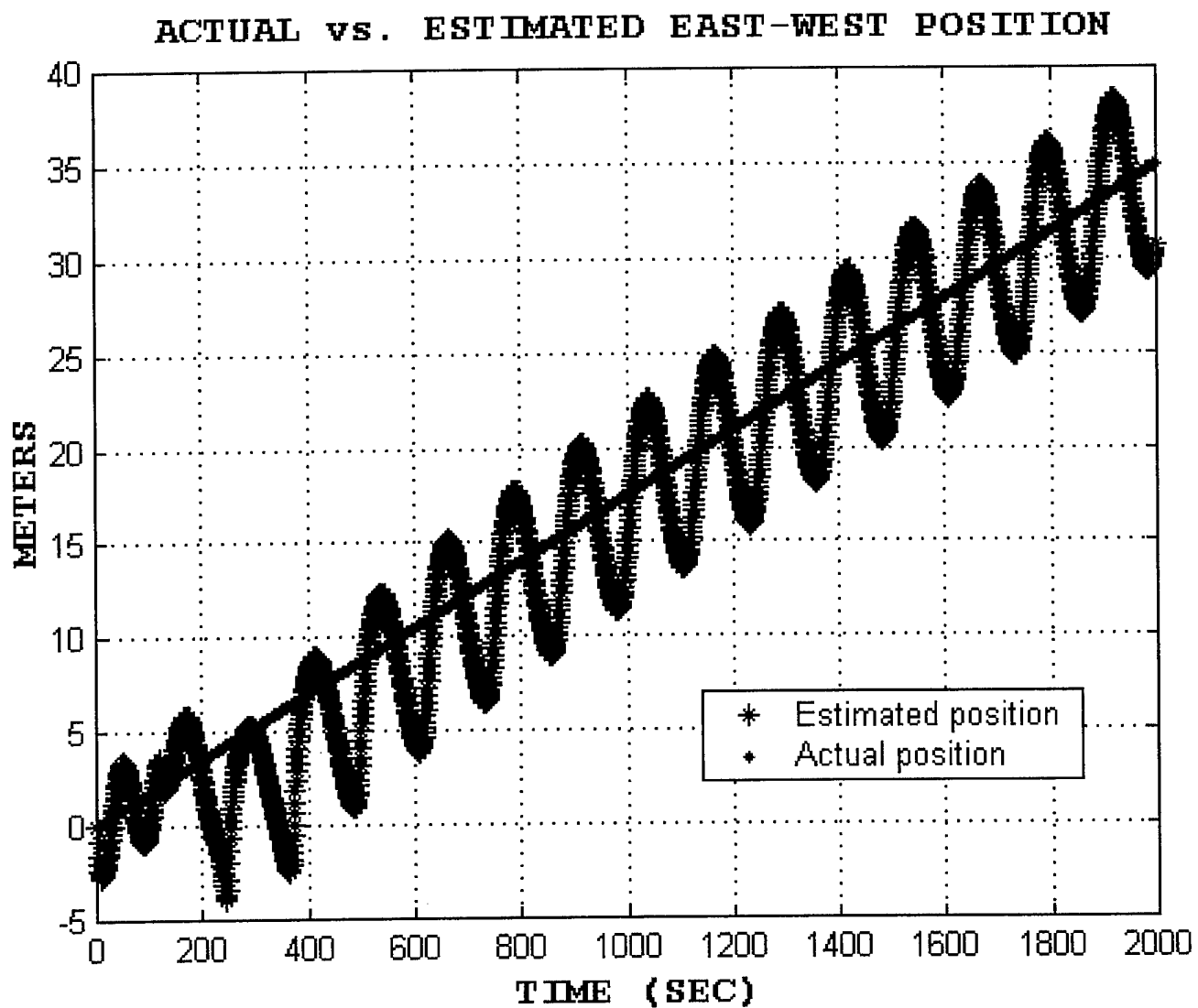


Figure 4.7 – East-west EKF position estimates vs. actual position for 30 meter “master” AUV path radius

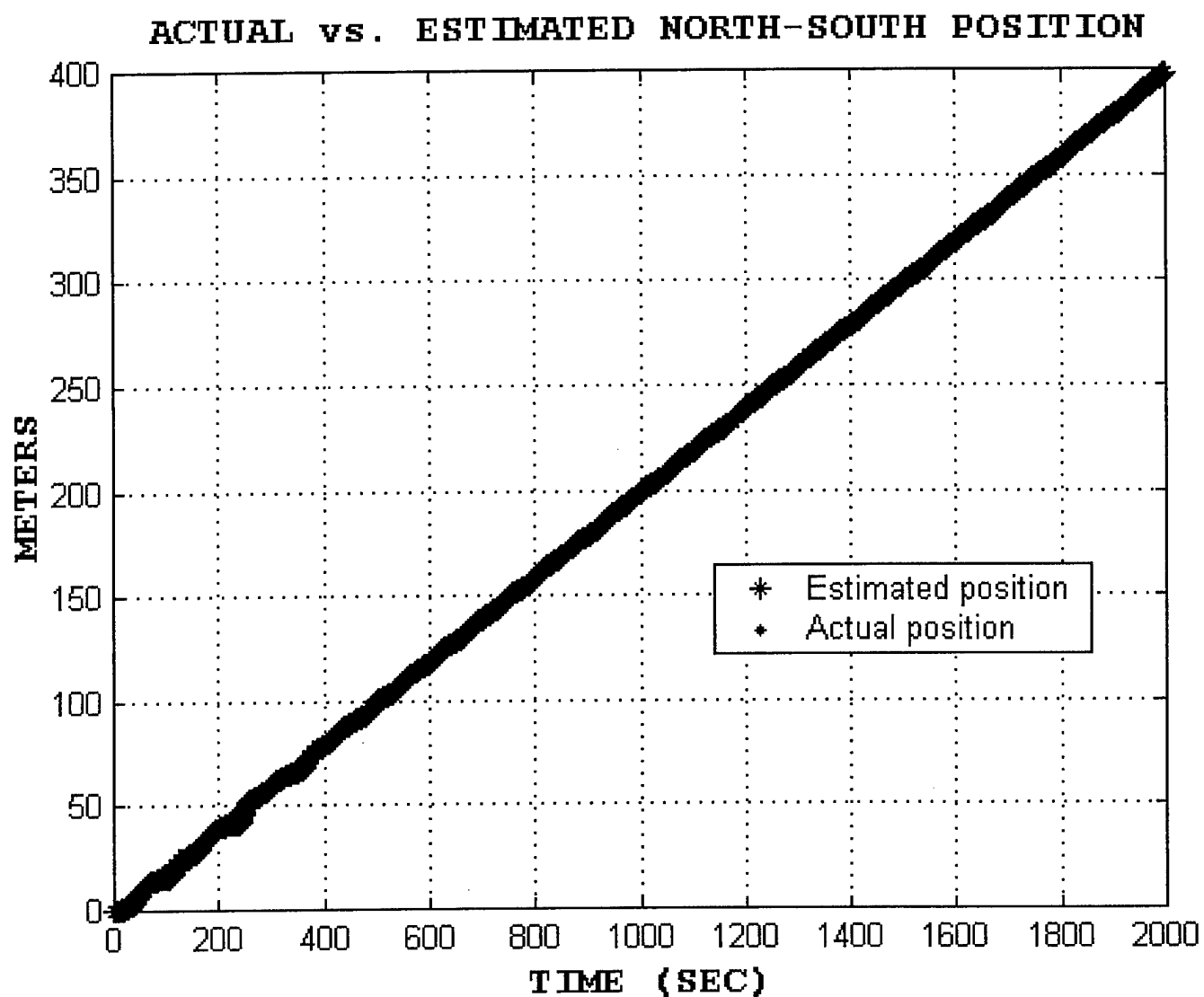


Figure 4.8 – North-south EKF position estimates vs. actual position for 30 meter
“master” AUV path radius

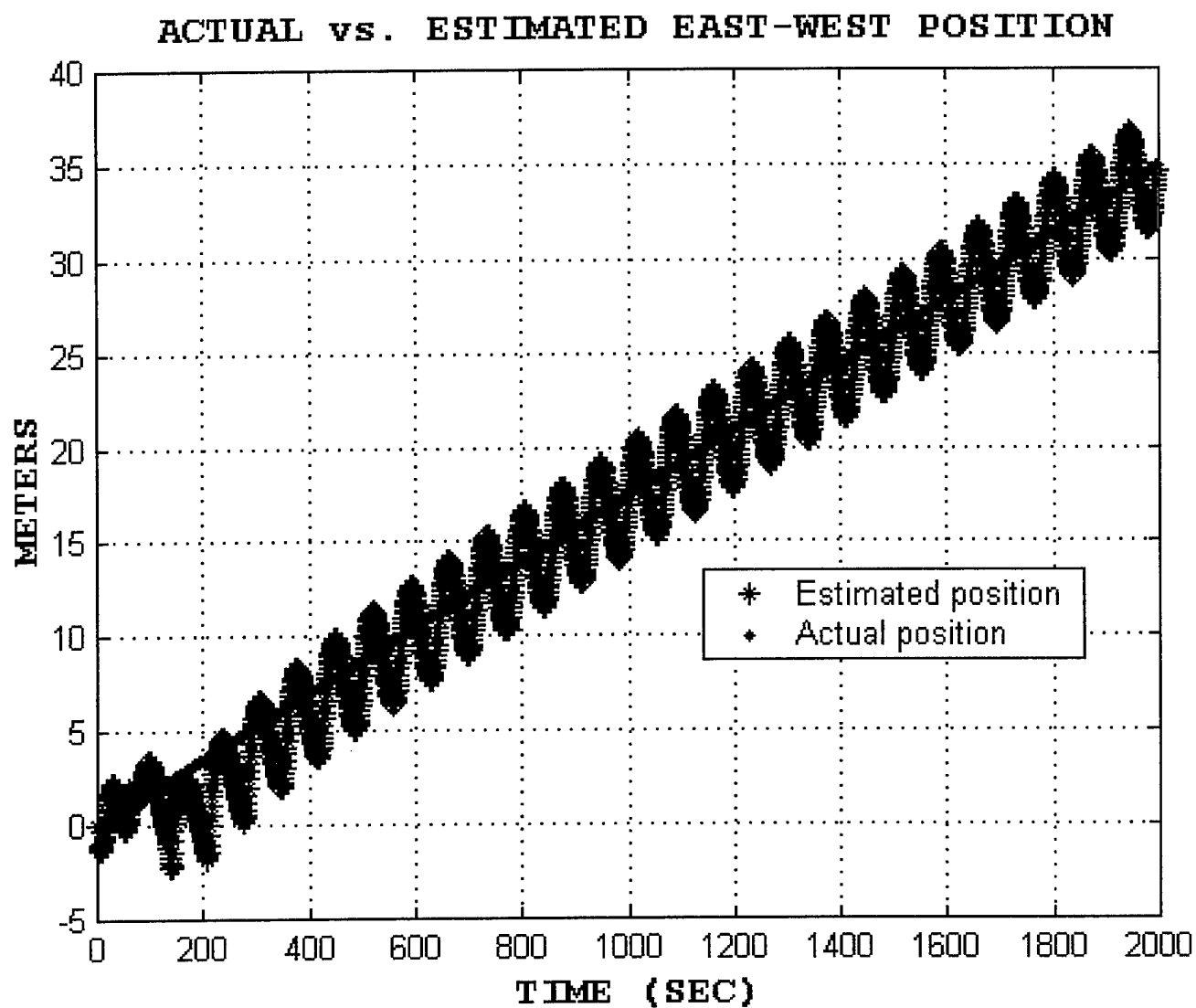


Figure 4.9 – East-west EKF position estimates vs. actual position for 17 meter
“master” AUV path radius

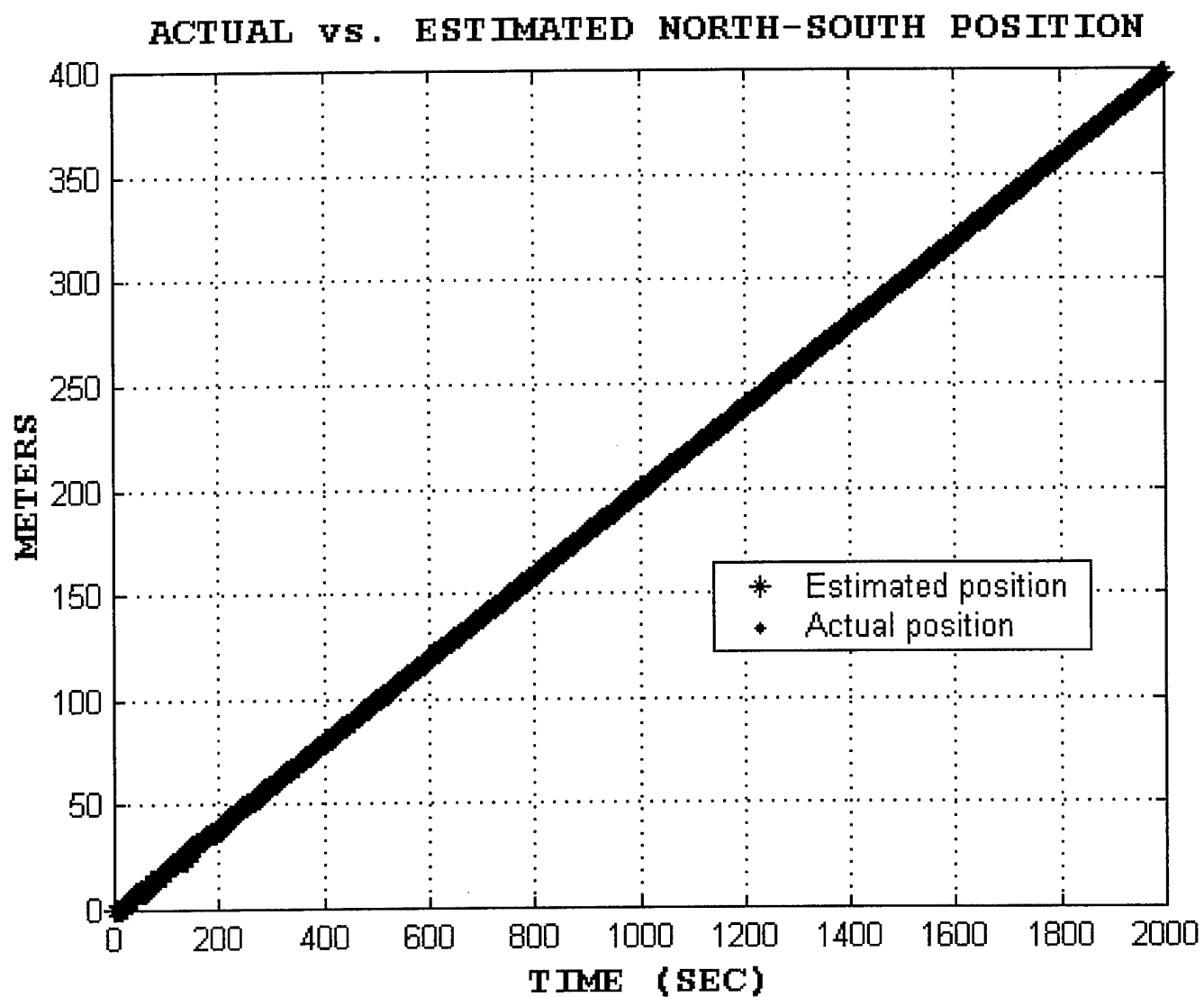


Figure 4.10 – North-south EKF position estimates vs. actual position for 17 meter
“master” AUV path radius

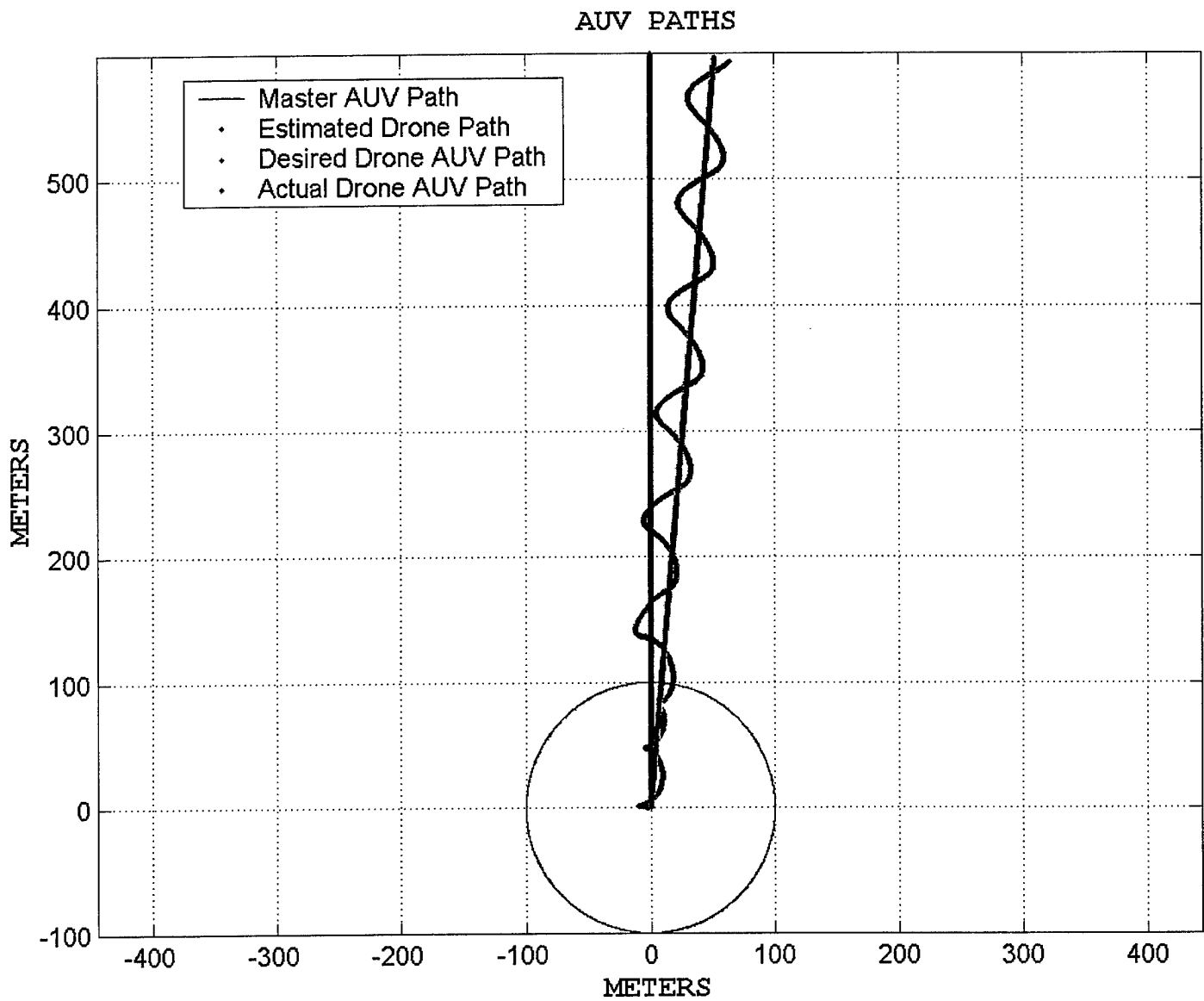


Figure 4.11 – AUV paths over 3000 seconds

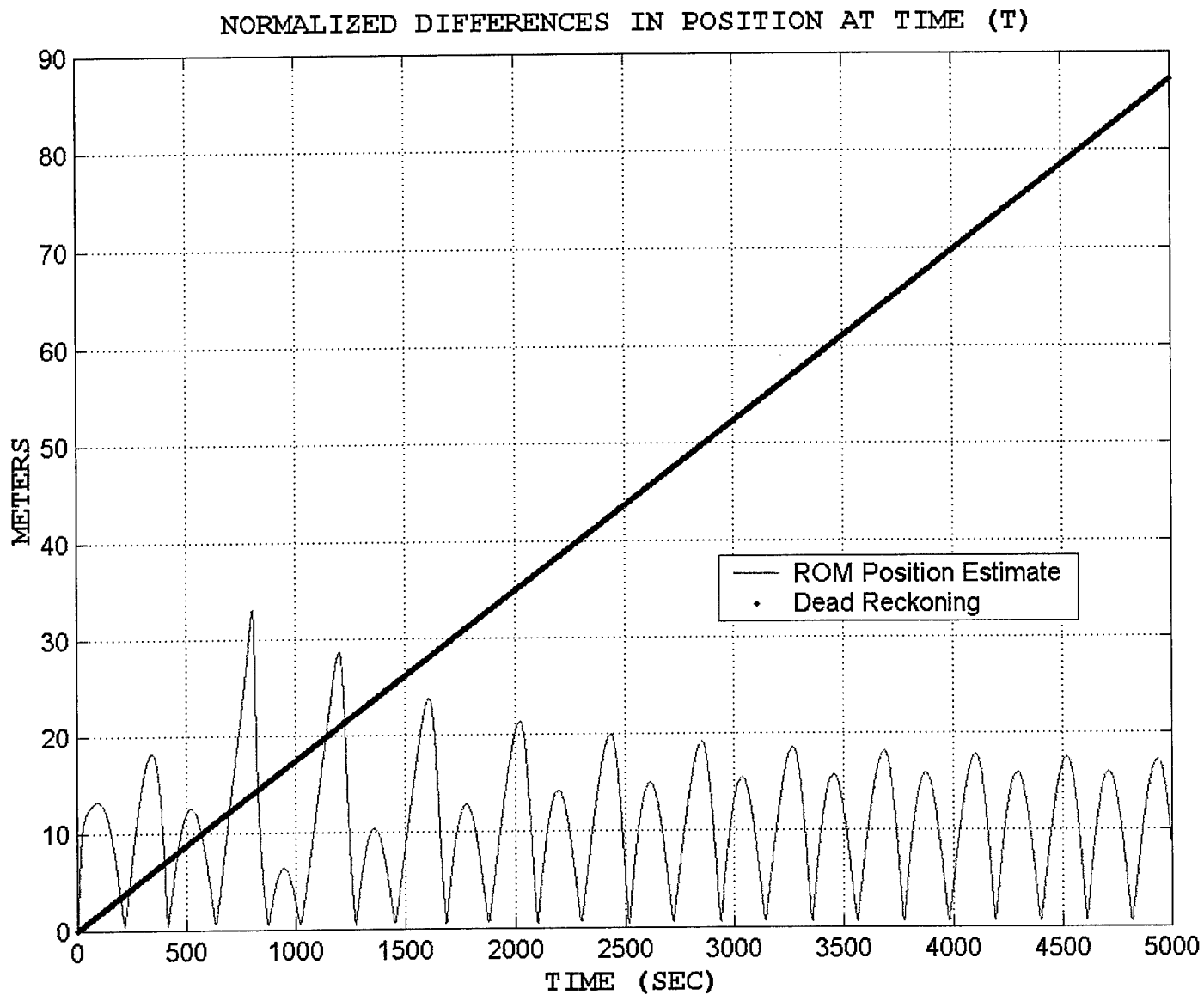


Figure 4.12 – Normalized differences between indicated and actual position for 100 meter “master” AUV path radius

DIFFERENCE BETWEEN ACTUAL AND ESTIMATED VALUES

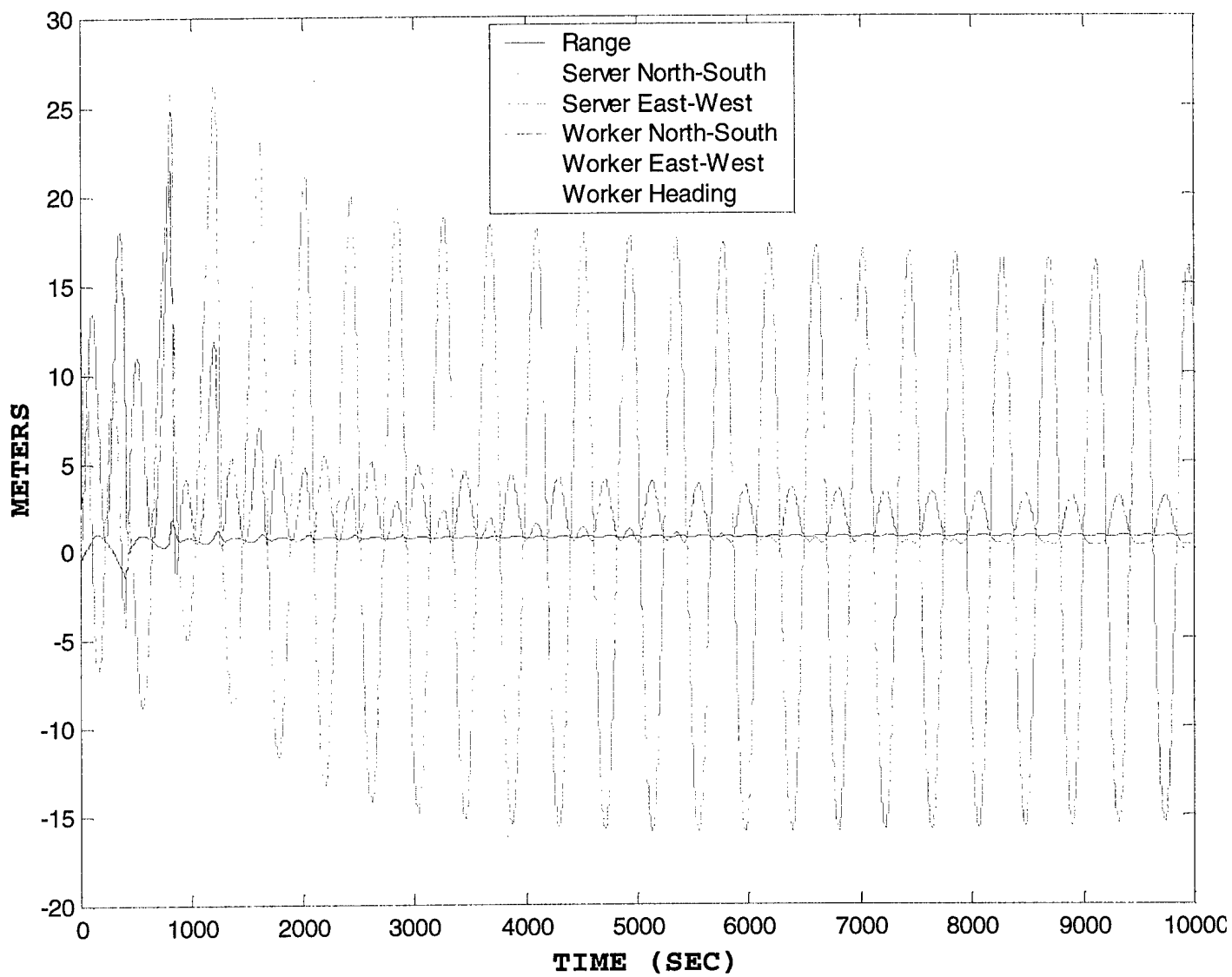


Figure 4.13 – Difference between actual and estimated position states for 100 meter “master” AUV path radius

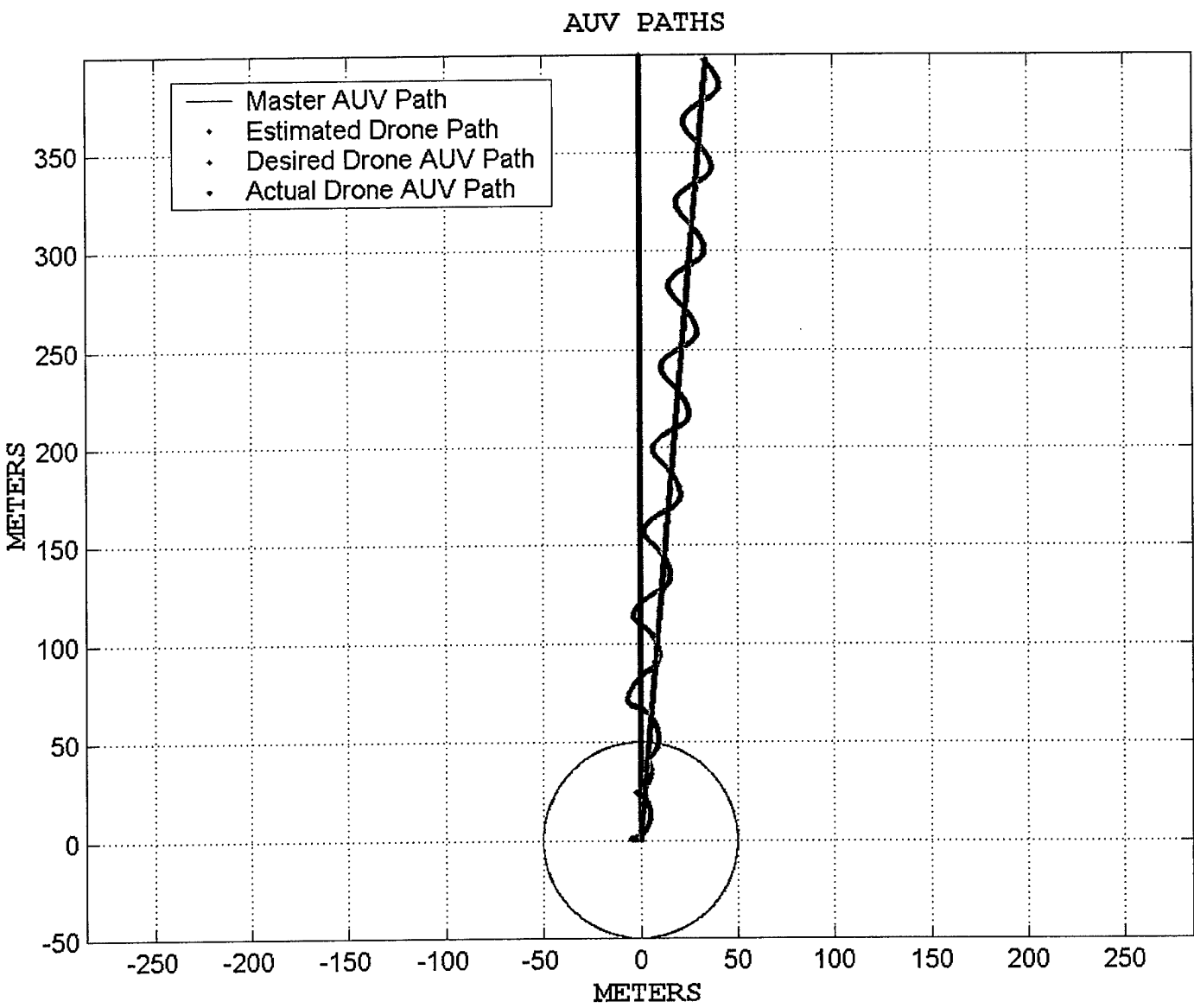


Figure 4.14 – AUV paths over 2000 seconds

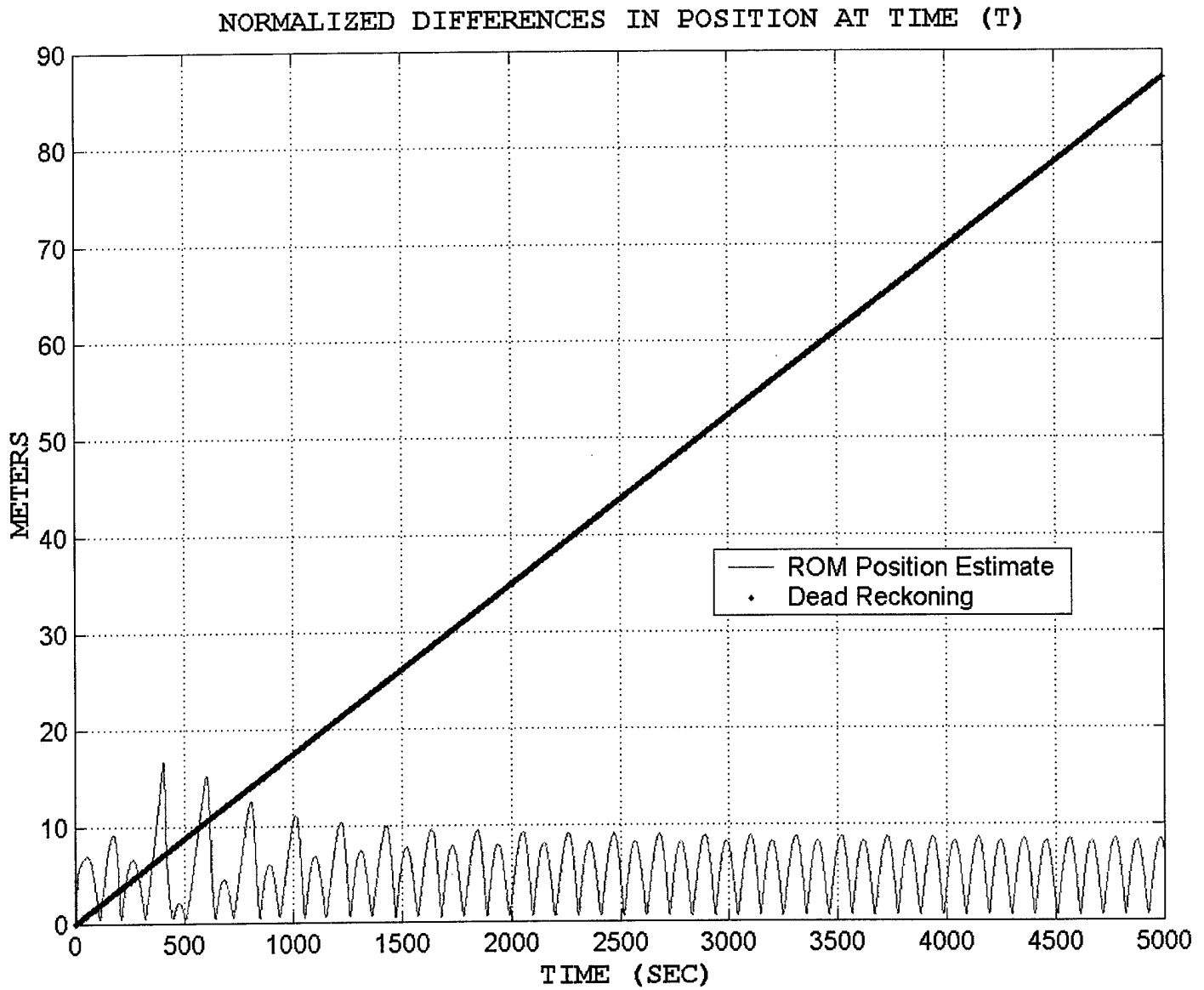


Figure 4.15 – Normalized differences between indicated and actual position for 50 meter “master” AUV path radius

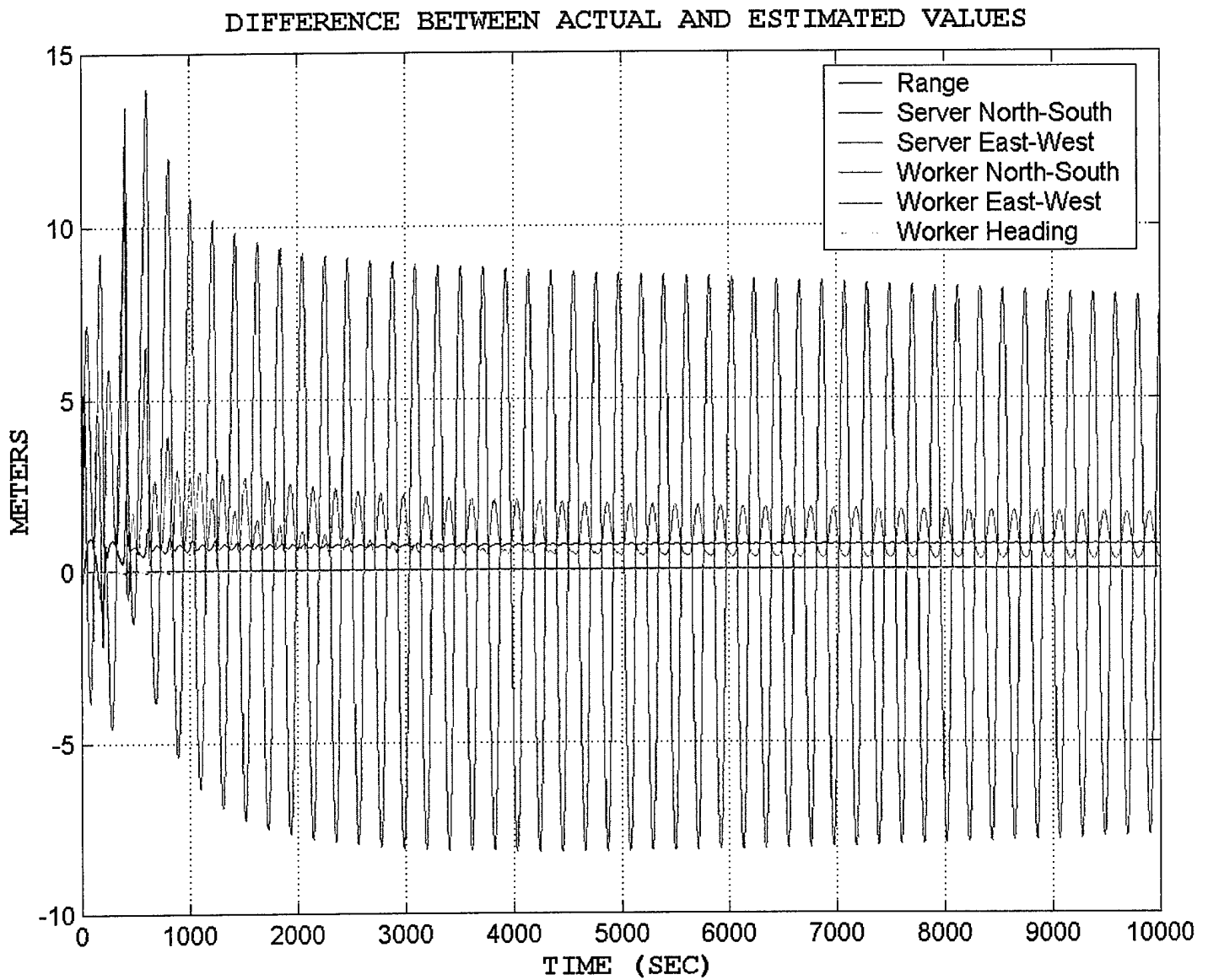


Figure 4.16 – Difference between actual and estimated position states for 50 meter
“master” AUV path radius

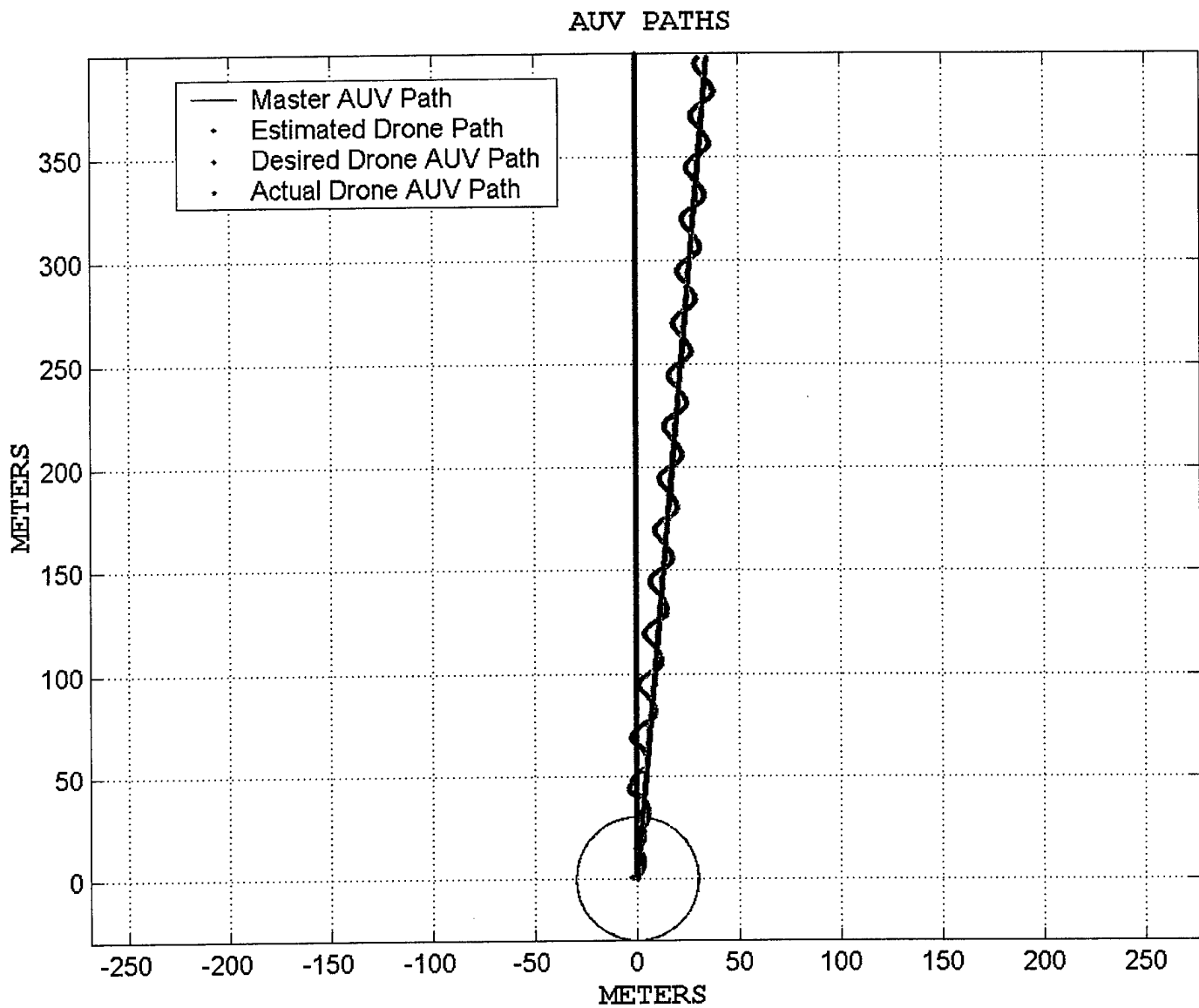


Figure 4.17 – AUV paths over 2000 seconds

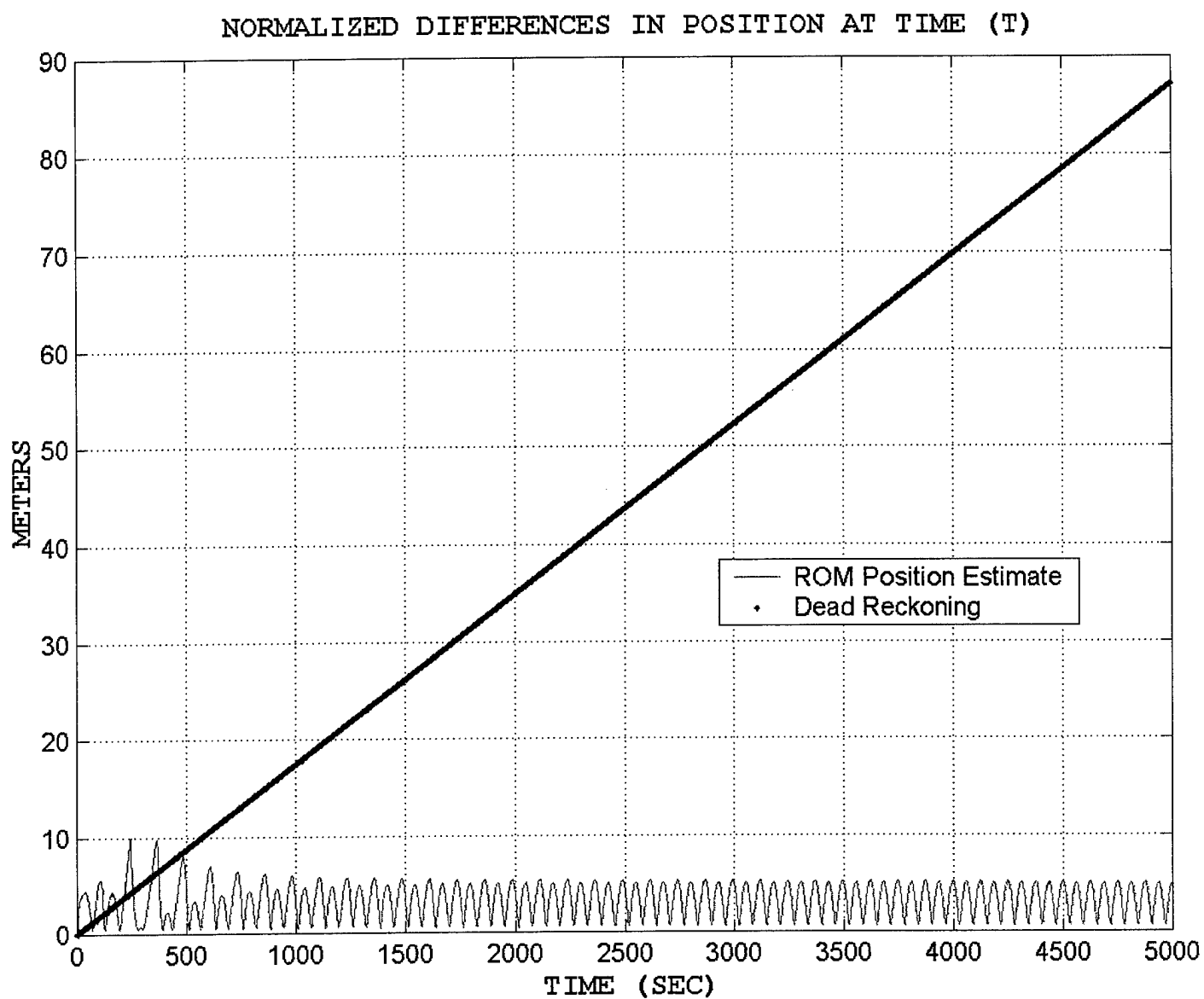
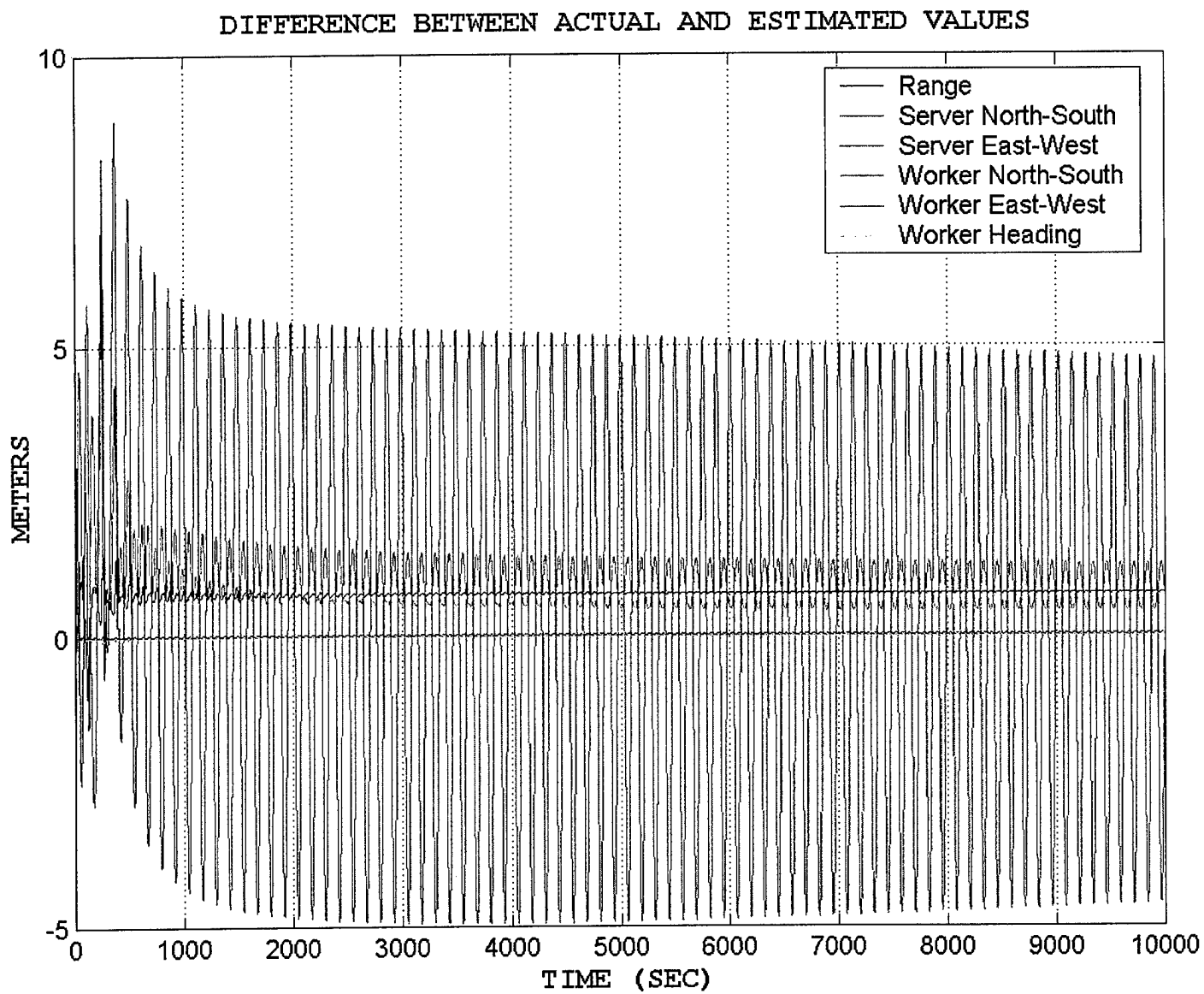


Figure 4.18 – Normalized differences between indicated and actual position for 30 meter “master” AUV path radius



**Figure 4.19 – Difference between actual and estimated position states for 30 meter
“master” AUV path radius**

AUV PATHS

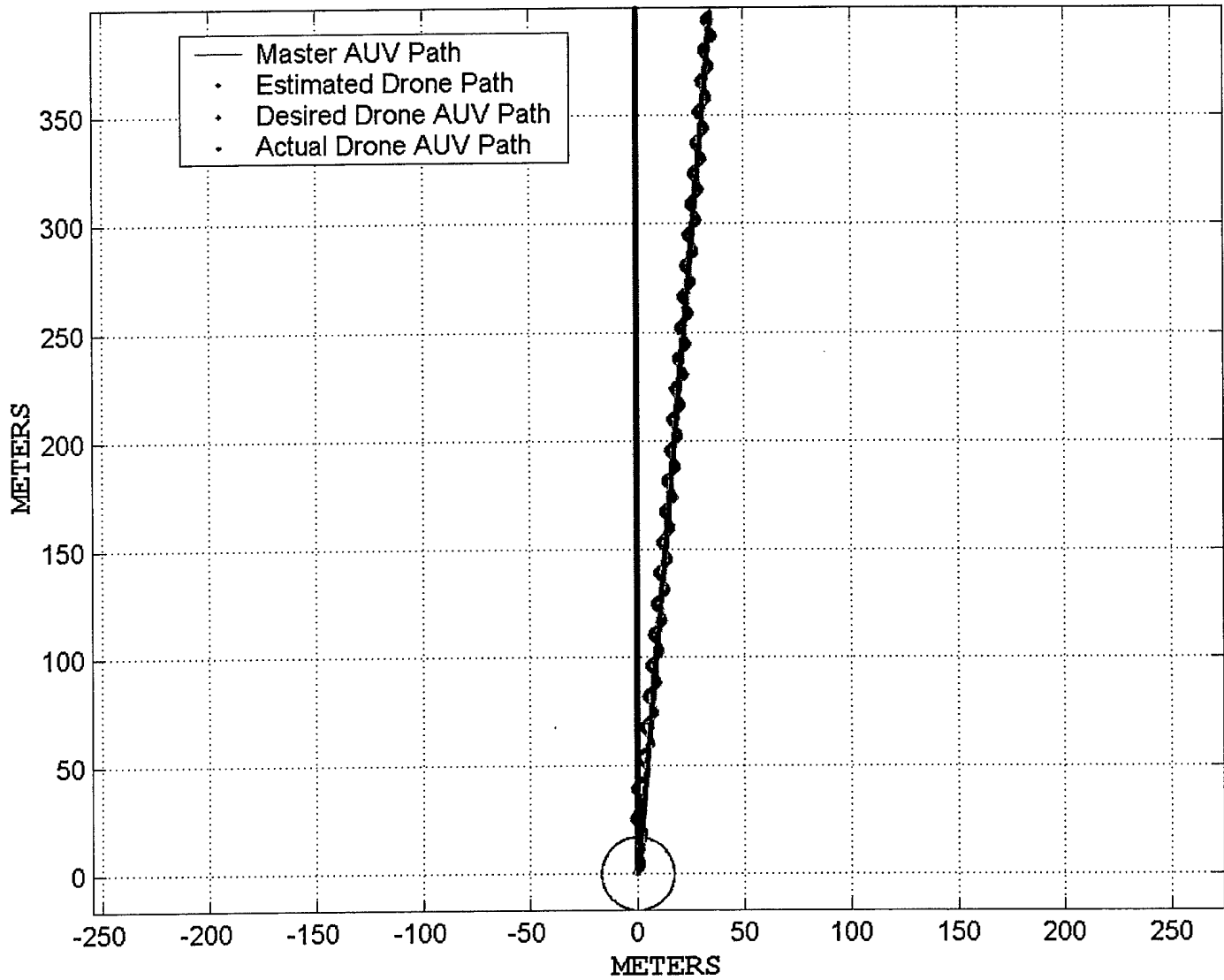


Figure 4.20 – AUV paths over 2000 seconds

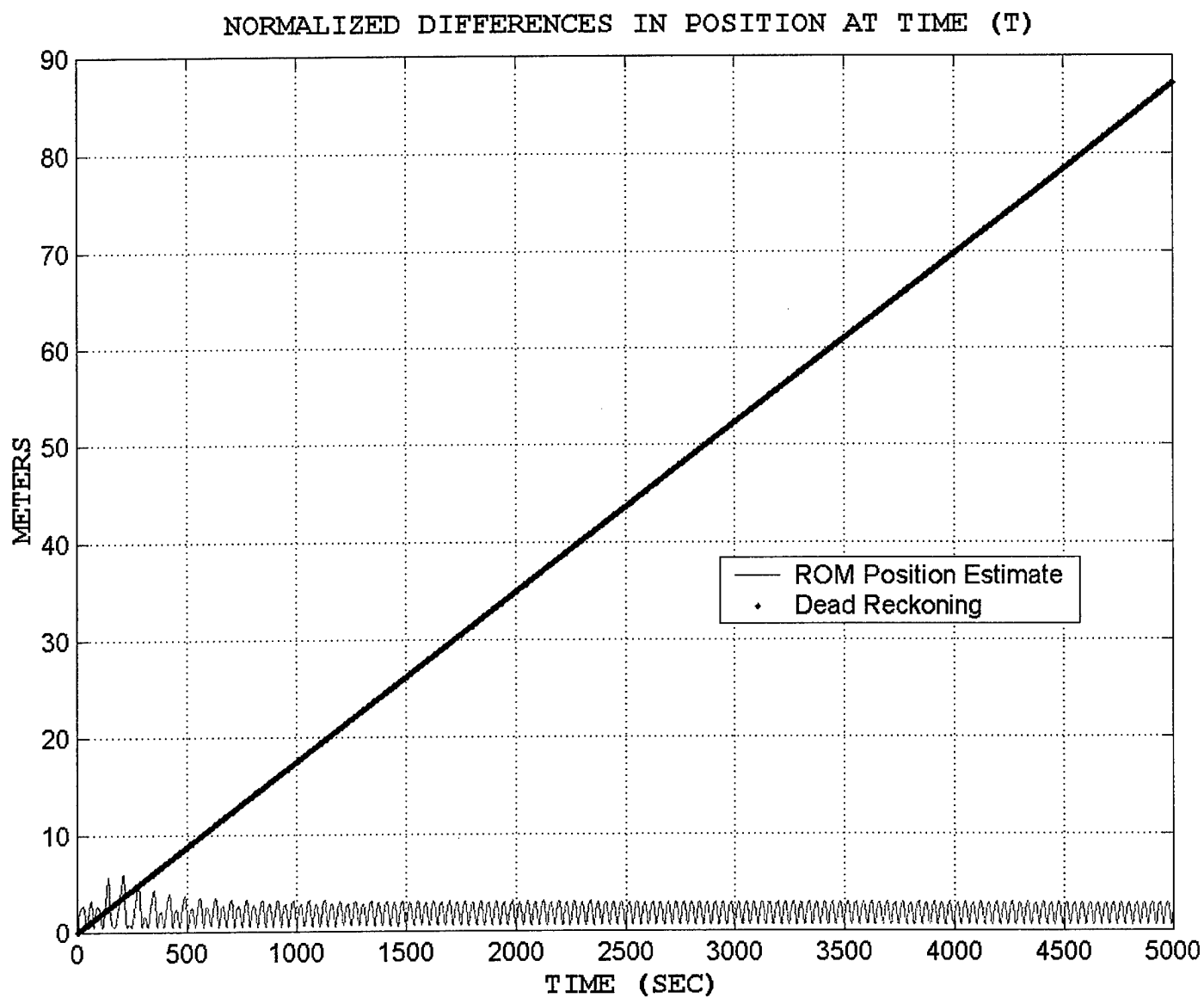


Figure 4.21 – Normalized differences between indicated and actual position for 17 meter “master” AUV path radius

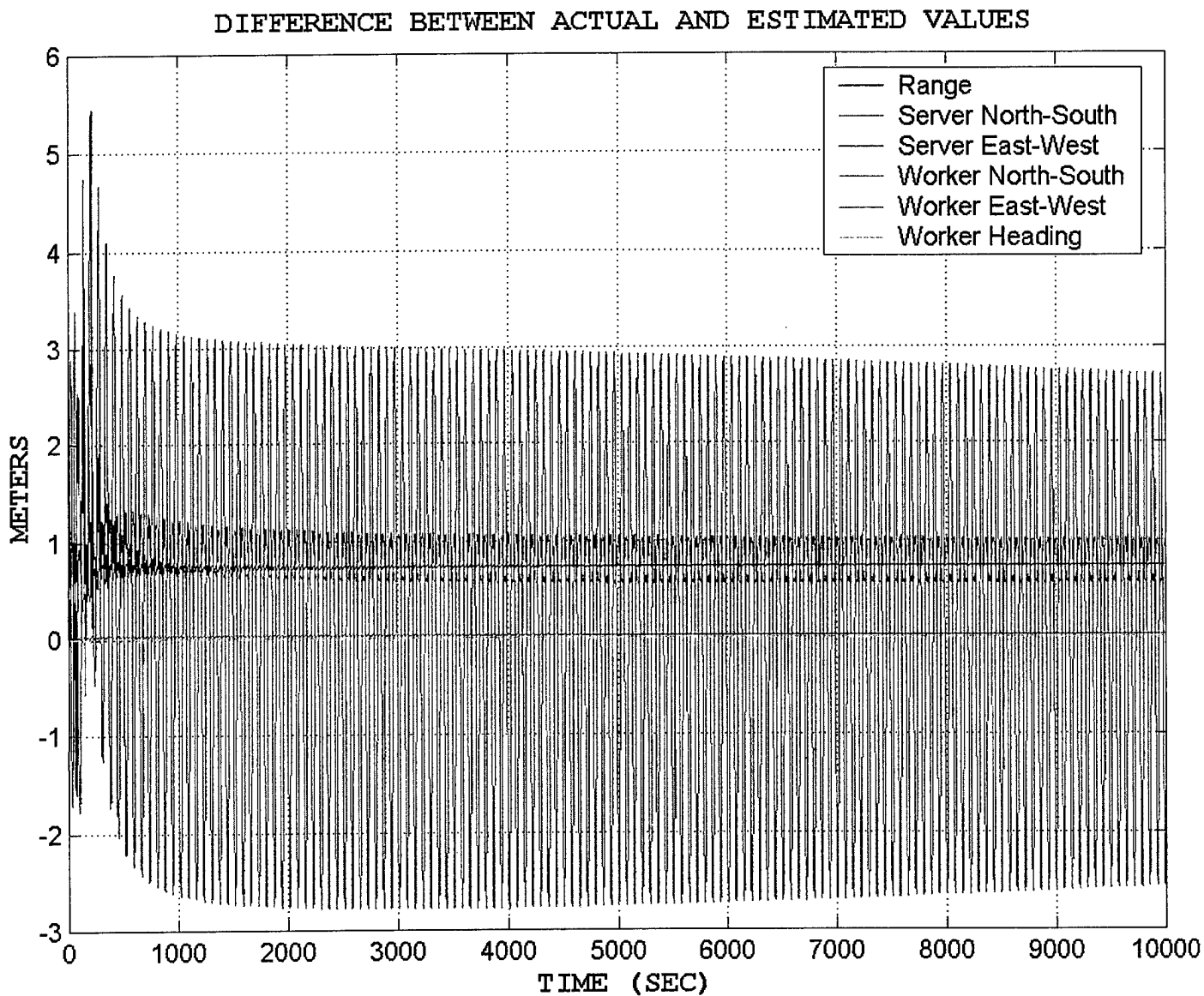


Figure 4.22 – Difference between actual and estimated position states for 17 meter
“master” AUV path radius

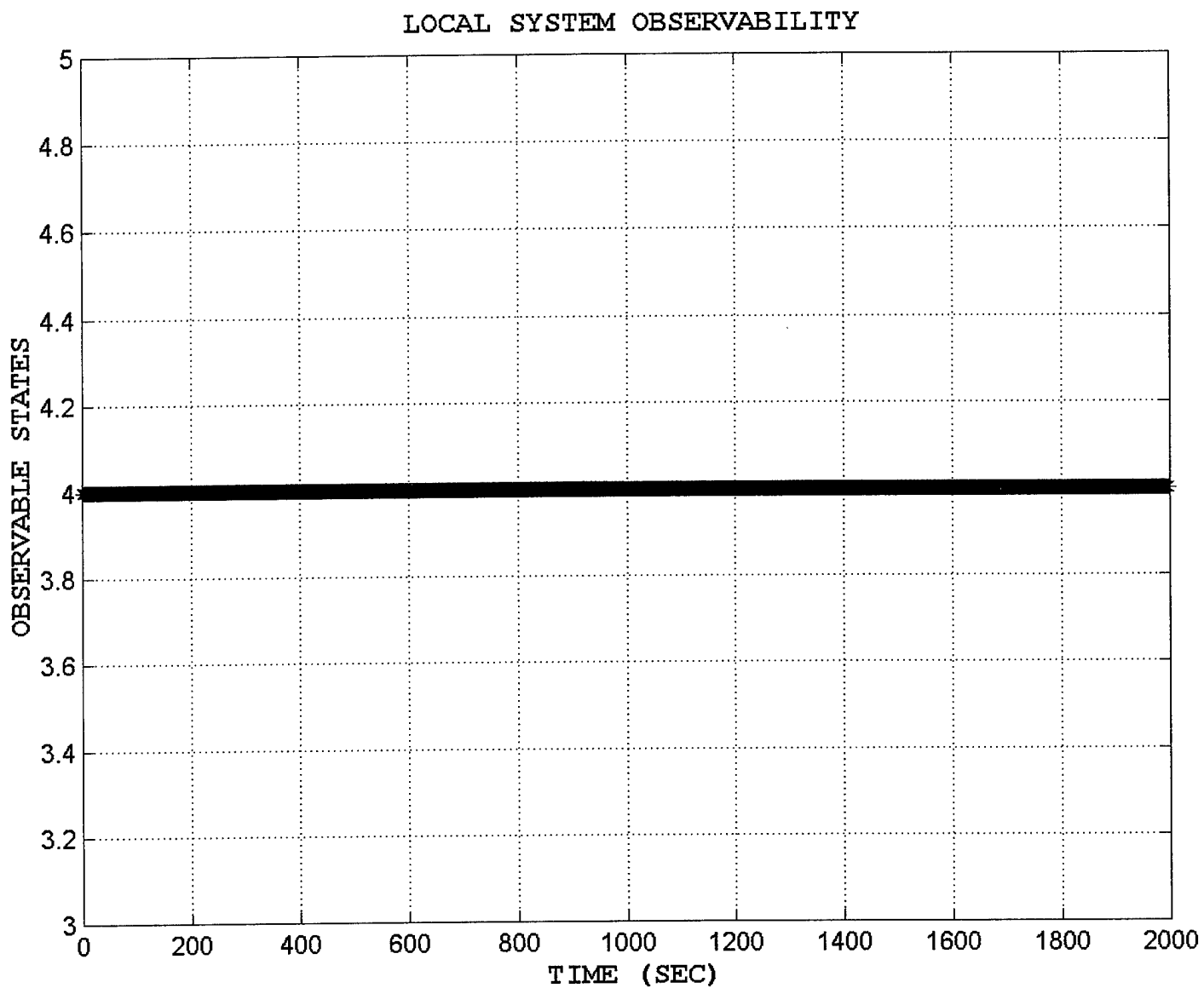


Figure 4.23 – Local Observability for all “master” AUV path radii with no initial drone condition data

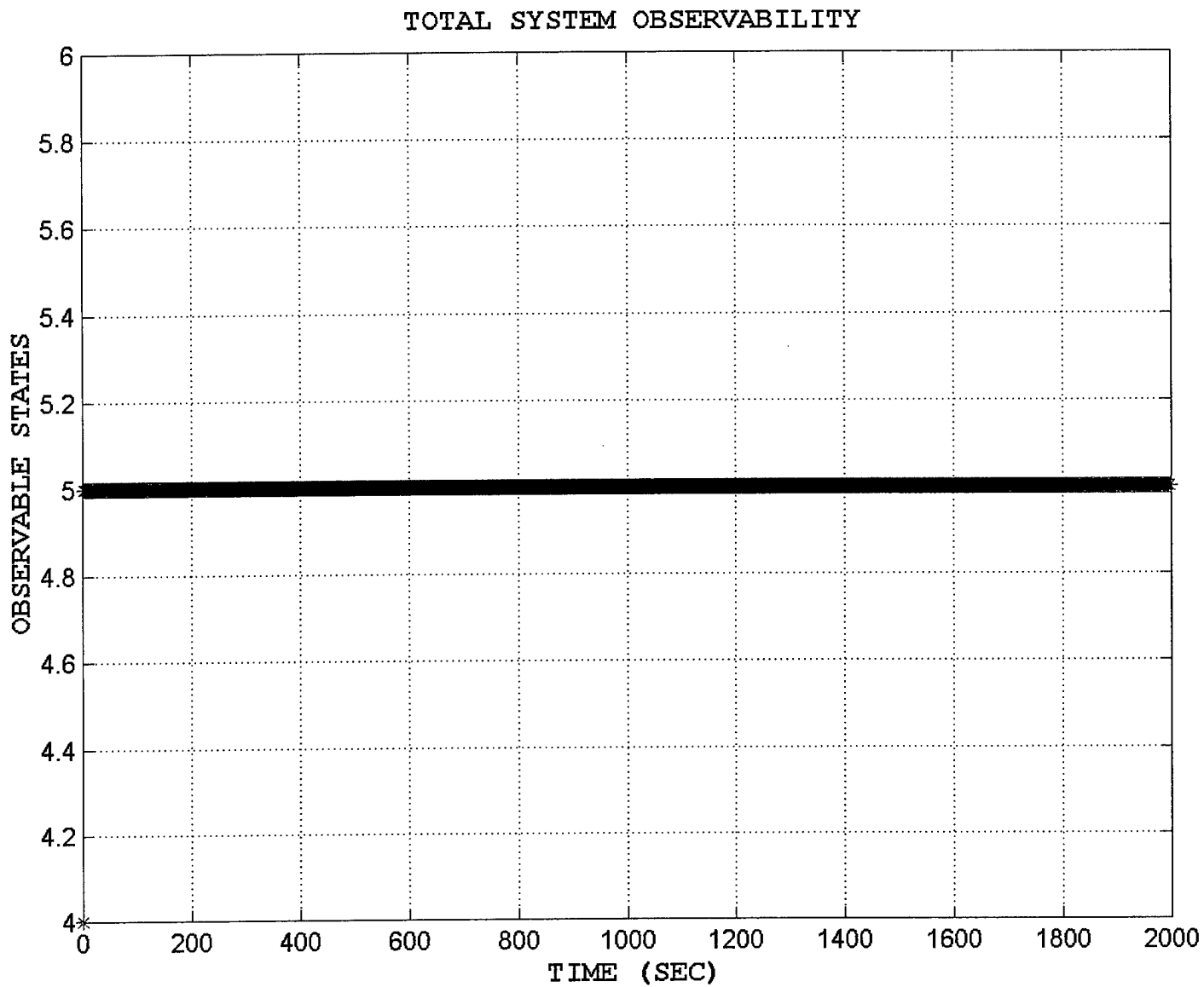


Figure 4.24 – Total observability for all “master” AUV path radii with no initial drone condition data

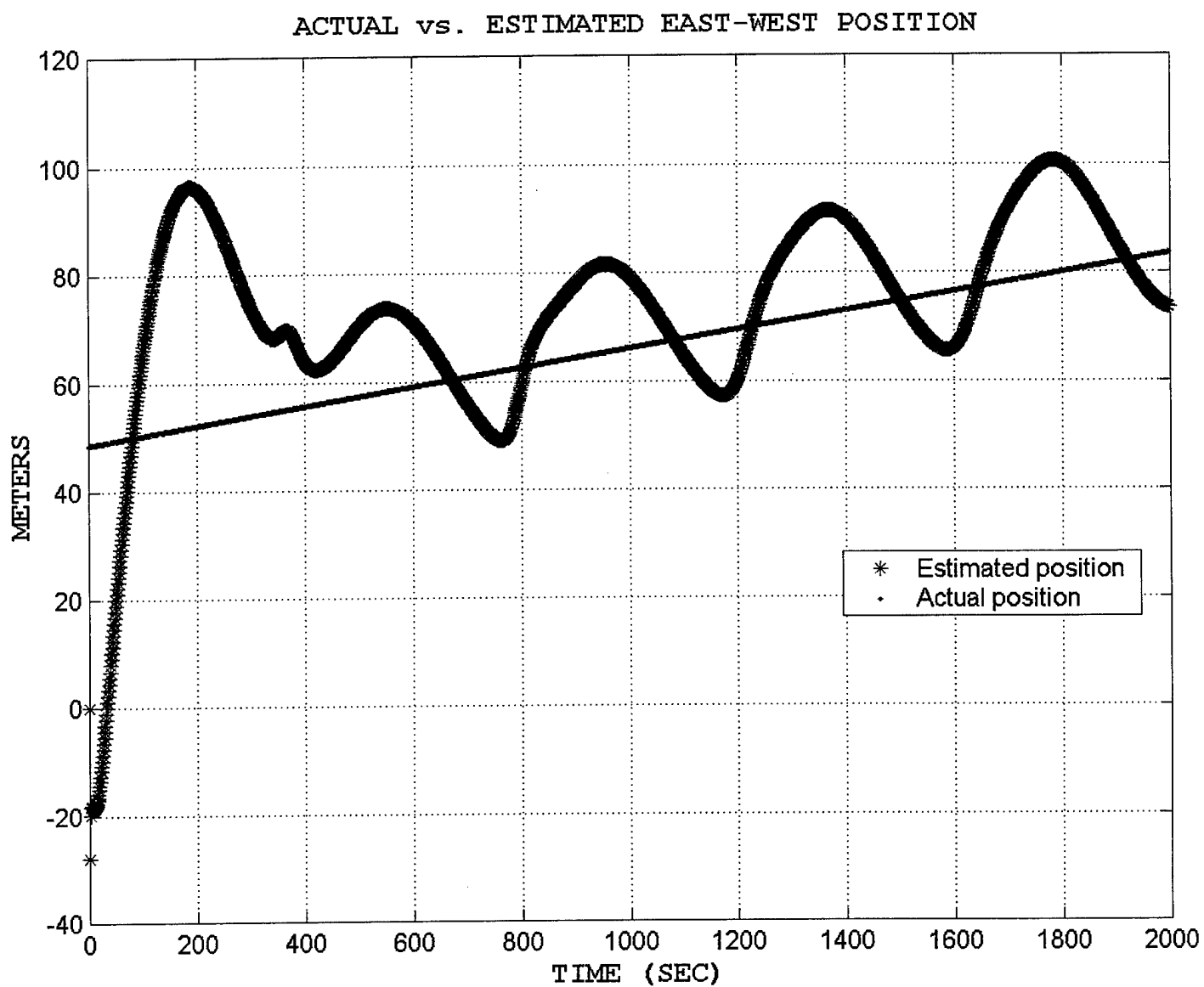


Figure 4.25 - East-west EKF position estimates vs. actual position for 100 meter “master” AUV path radius with no initial position data

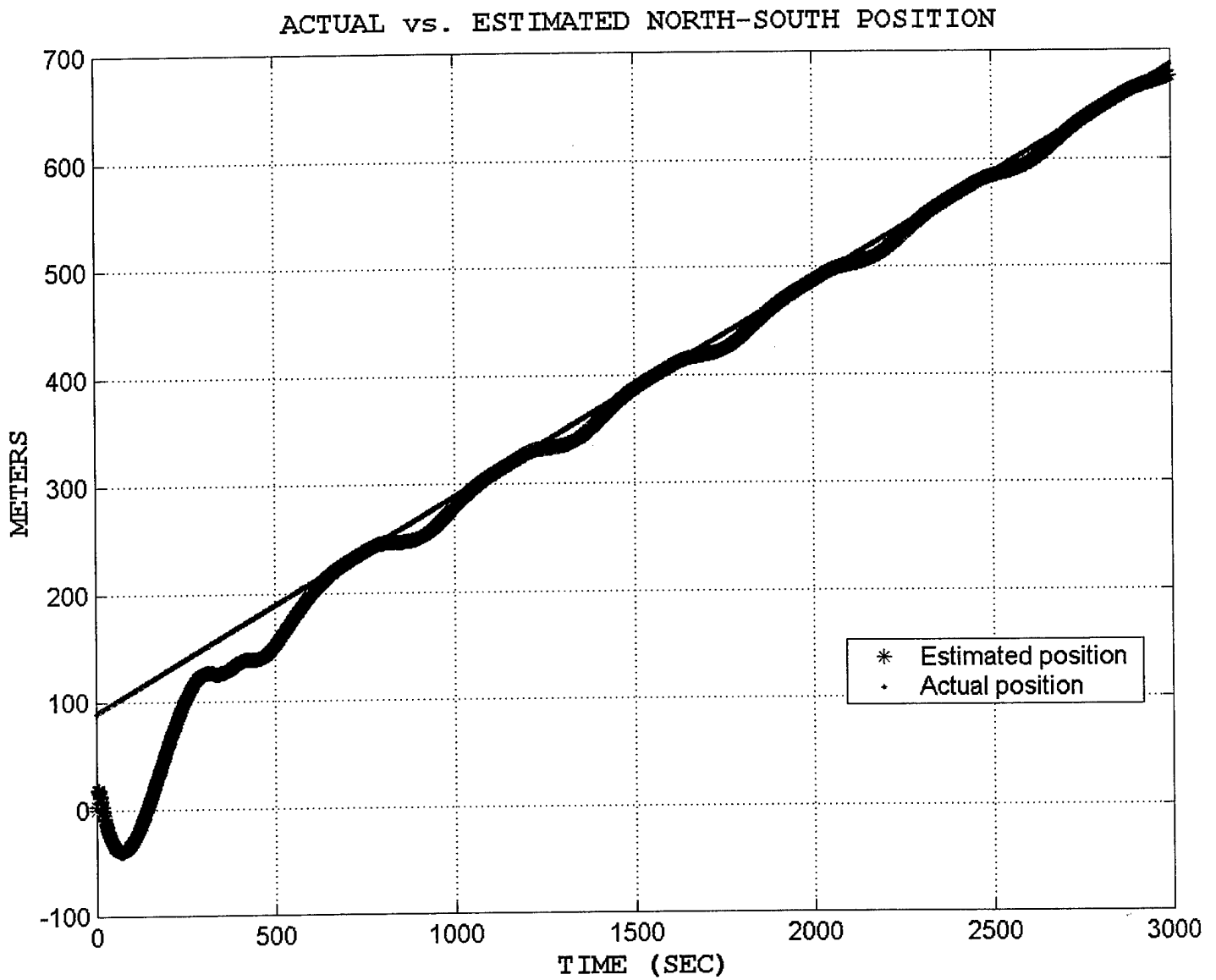


Figure 4.26 - North-south EKF position estimates vs. actual position for 100 meter “master” AUV path radius with no initial position data

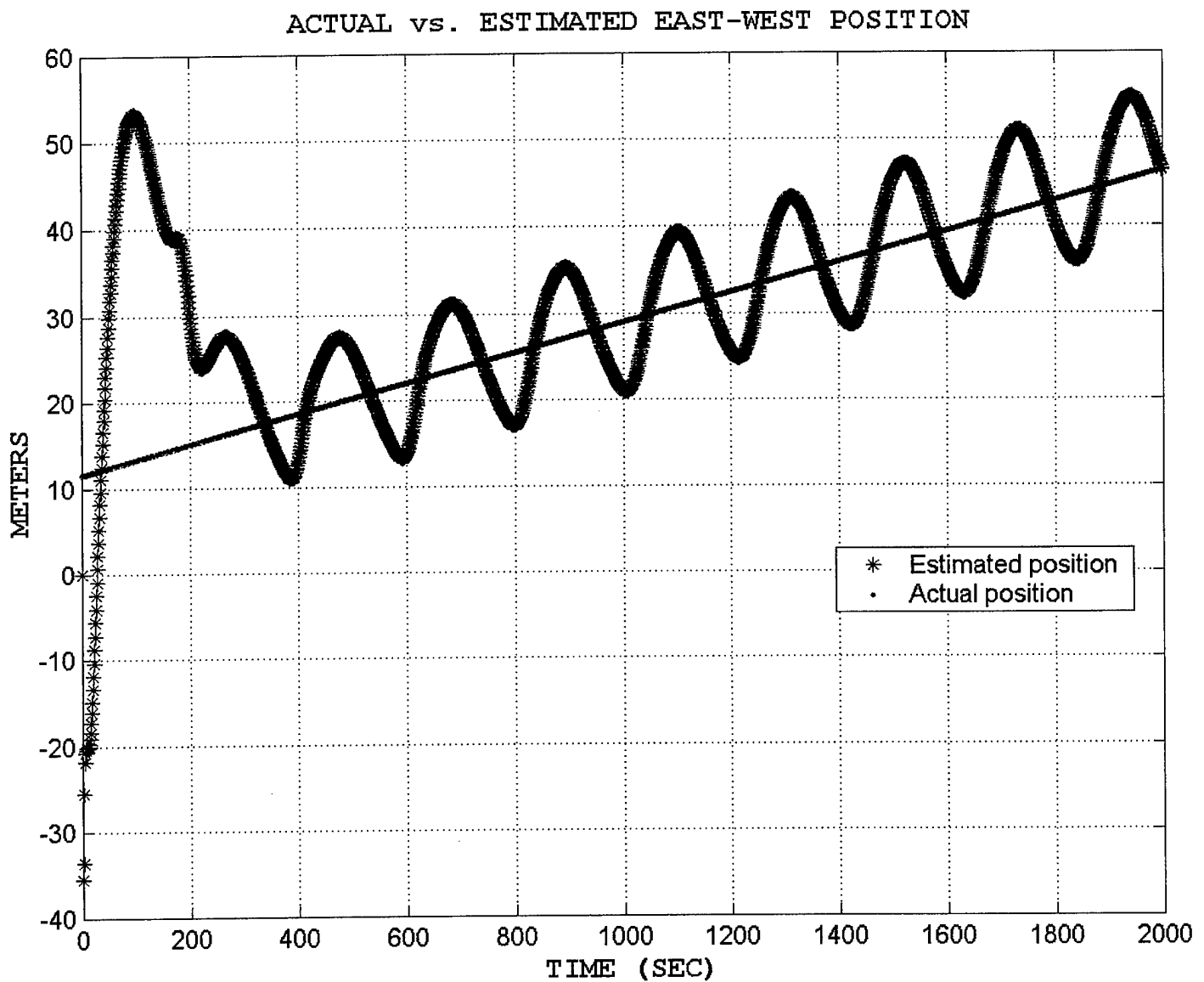


Figure 4.27 - East-west EKF position estimates vs. actual position for 50 meter “master” AUV path radius with no initial position data

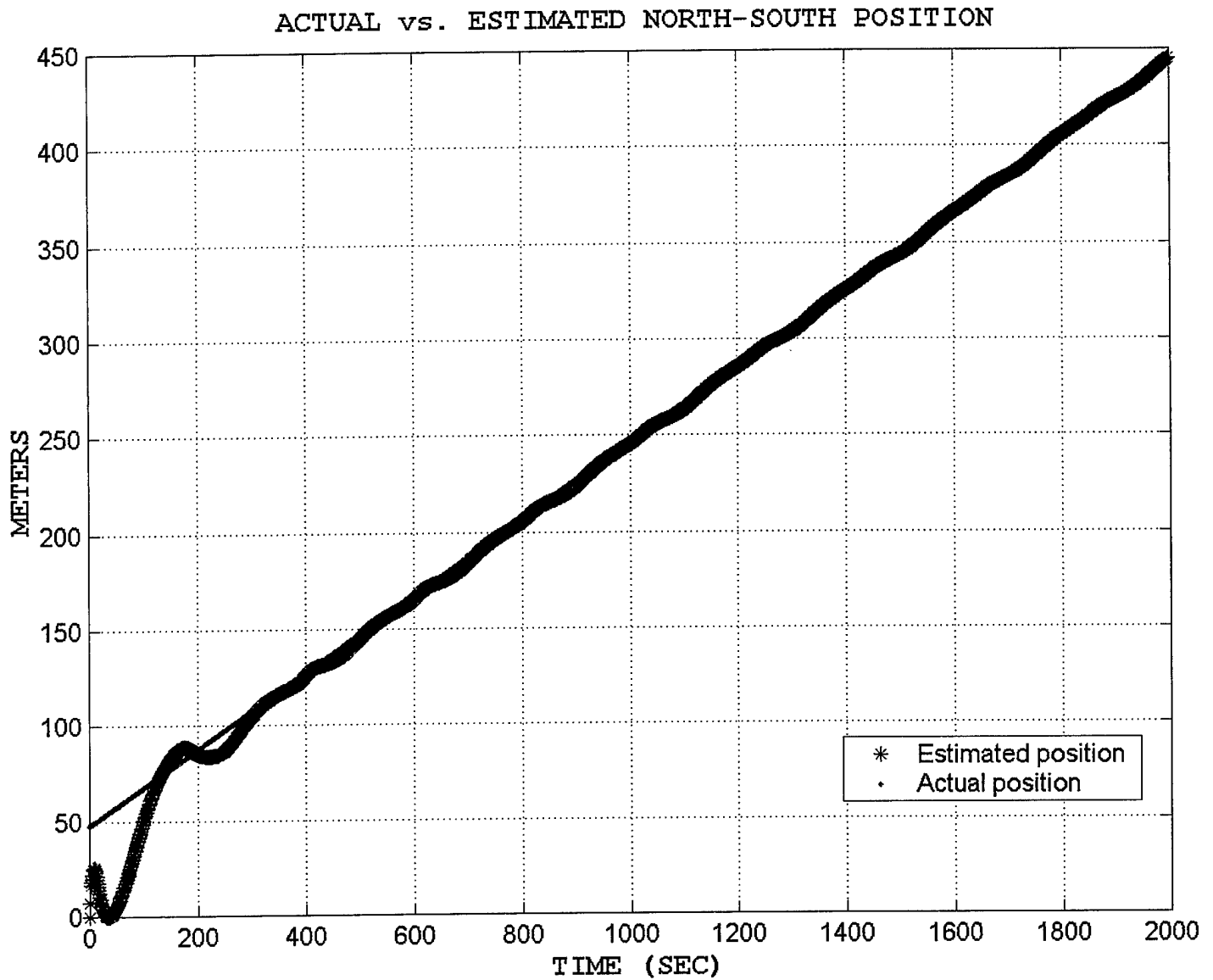


Figure 4.28 - North-south EKF position estimates vs. actual position for 50 meter “master” AUV path radius with no initial position data

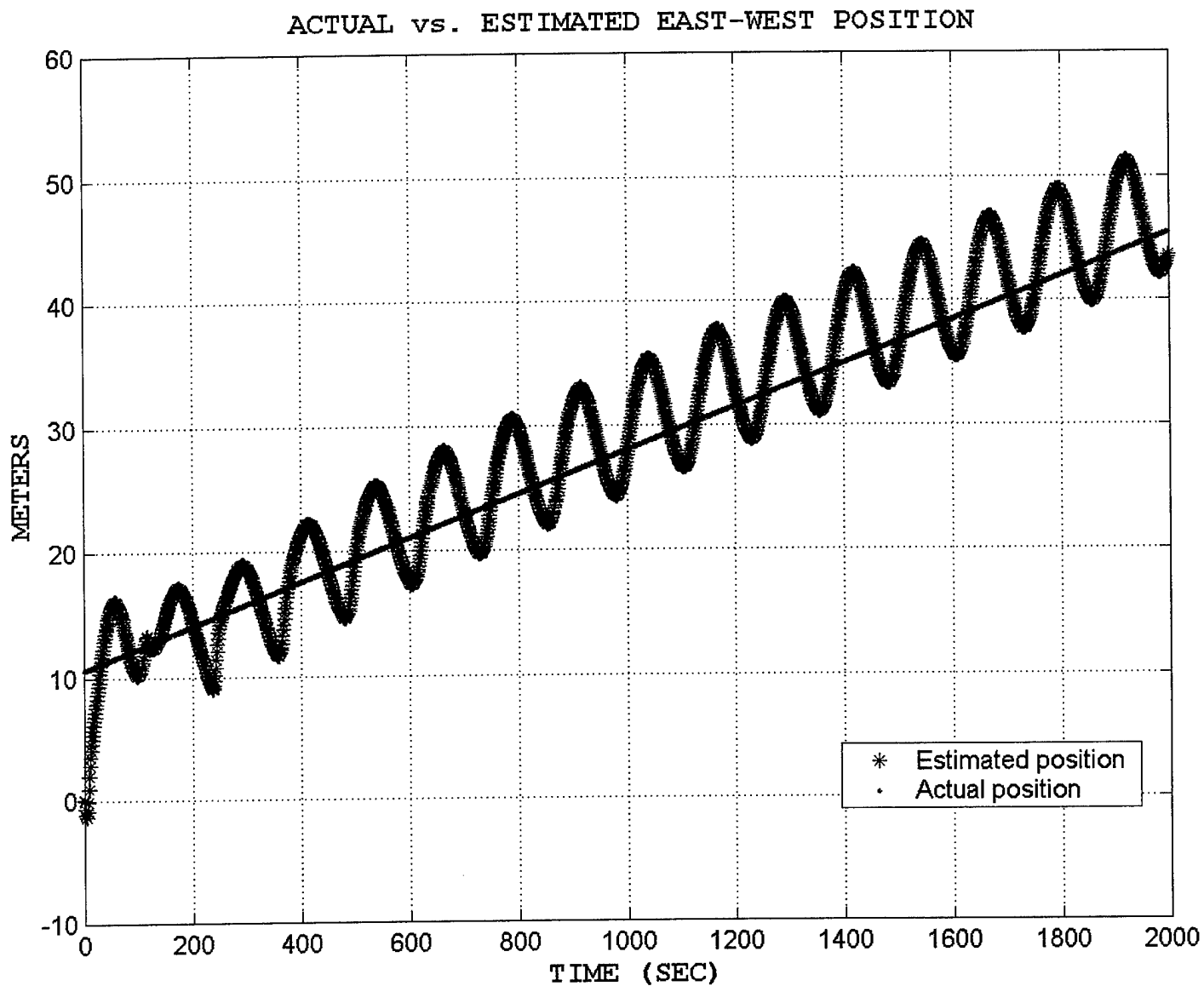


Figure 4.29 - East-west EKF position estimates vs. actual position for 30 meter "master" AUV path radius with no initial position data

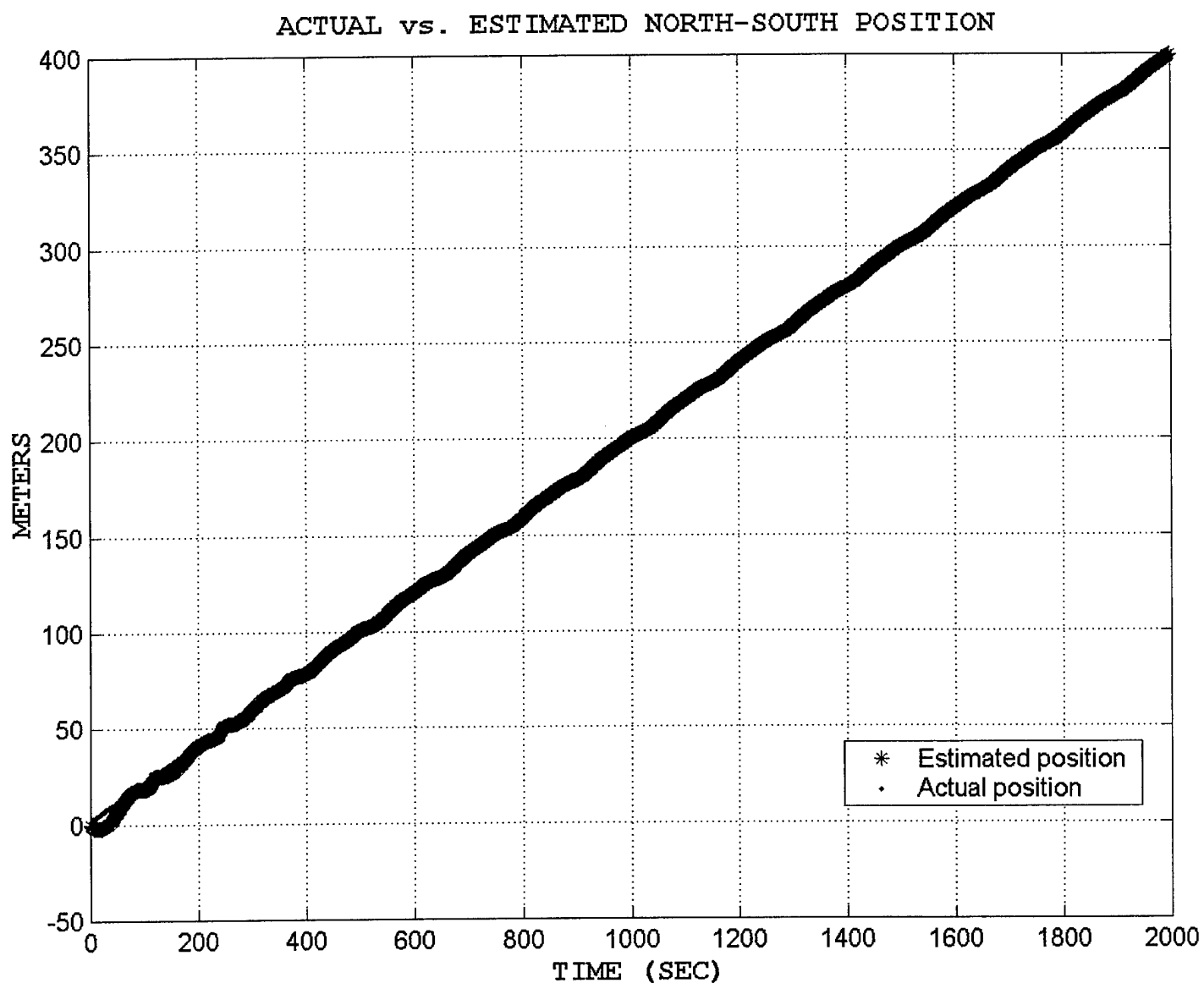


Figure 4.30 - North-south EKF position estimates vs. actual position for 30 meter “master” AUV path radius with no initial position data

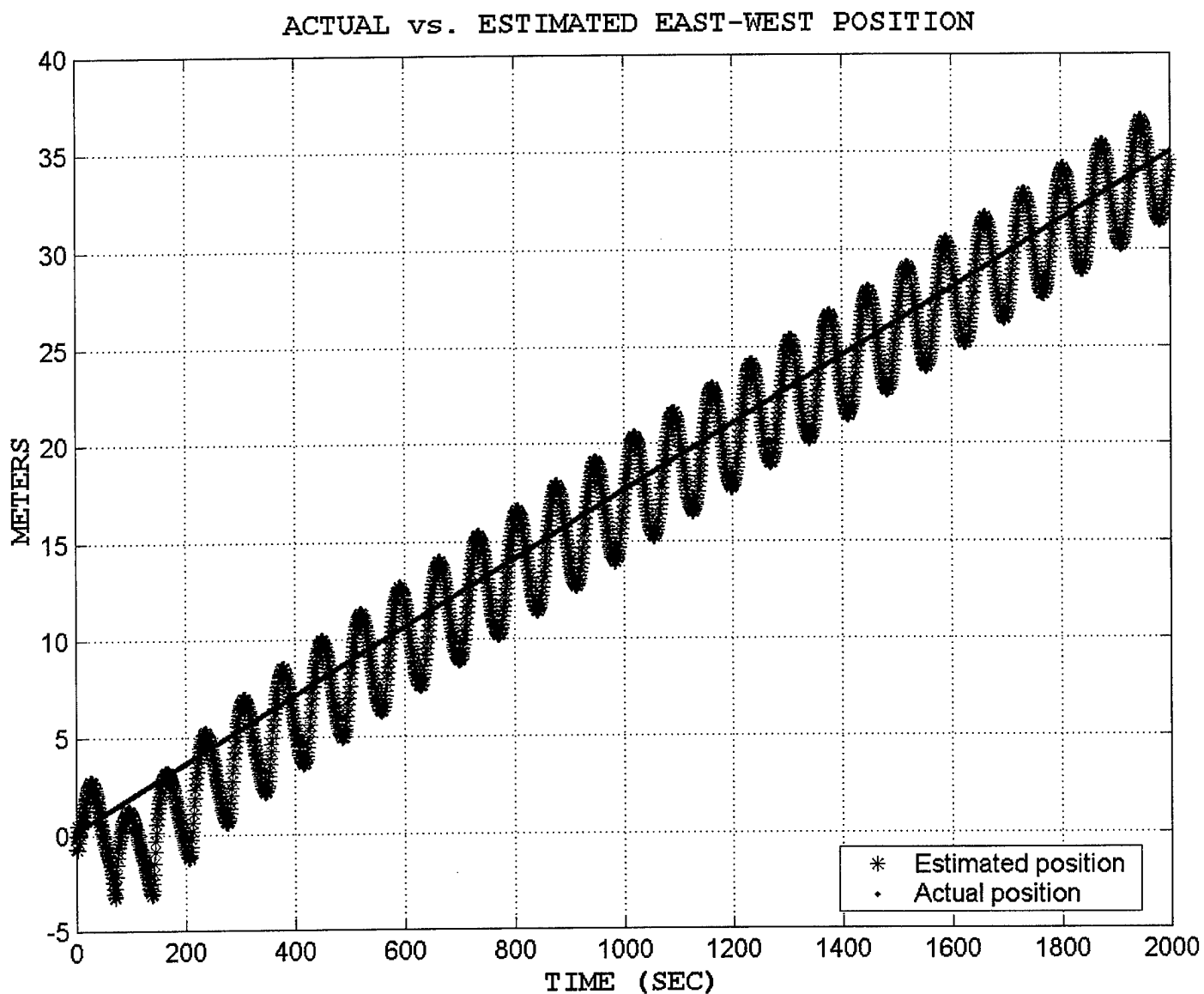


Figure 4.31 - East-west EKF position estimates vs. actual position for 17 meter “master” AUV path radius with no initial position data

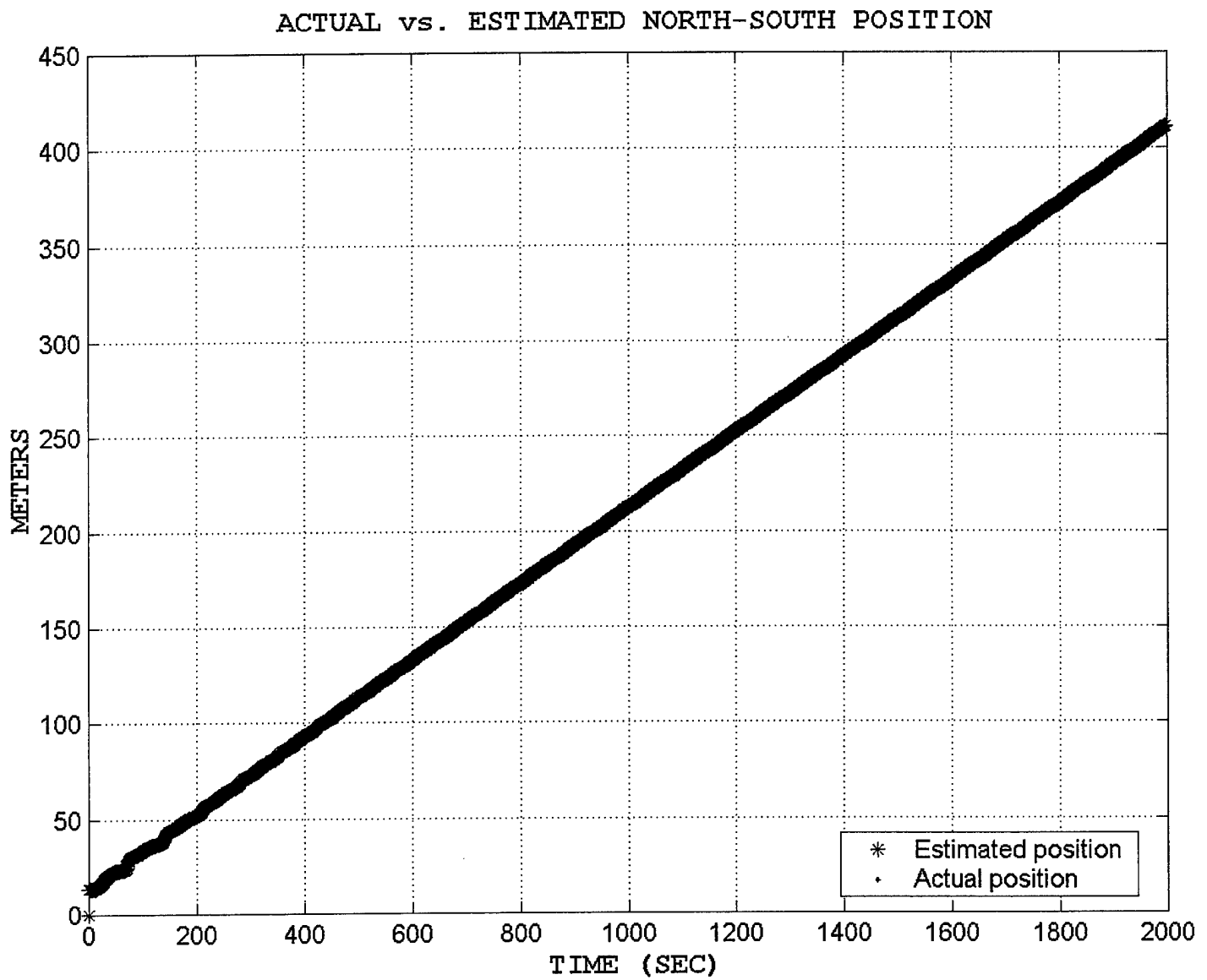


Figure 4.32 – North-south EKF position estimates vs. actual position for 17 meter “master” AUV path radius with no initial position data.

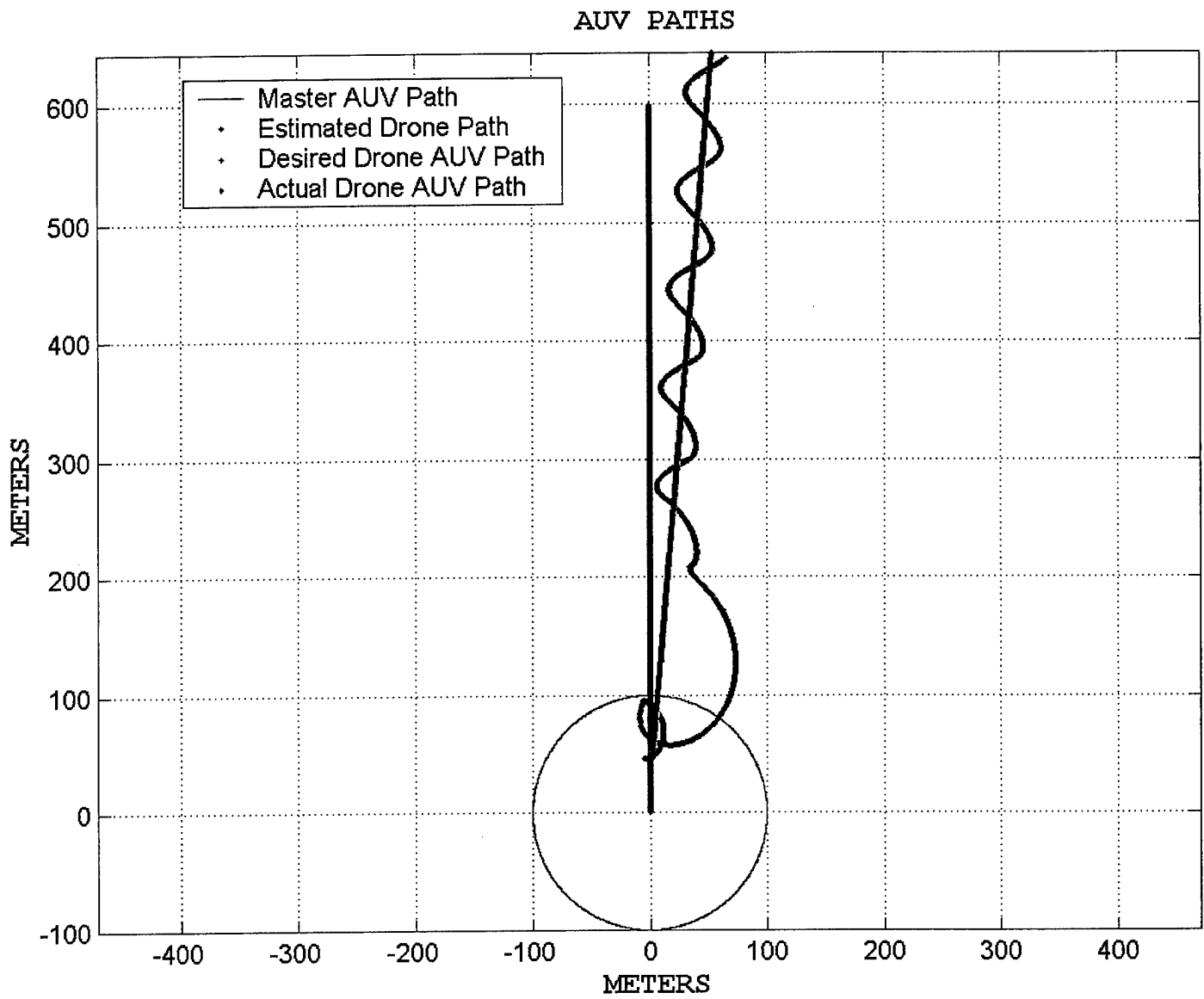


Figure 4.33 – AUV paths with no initial drone position data over 3000 seconds

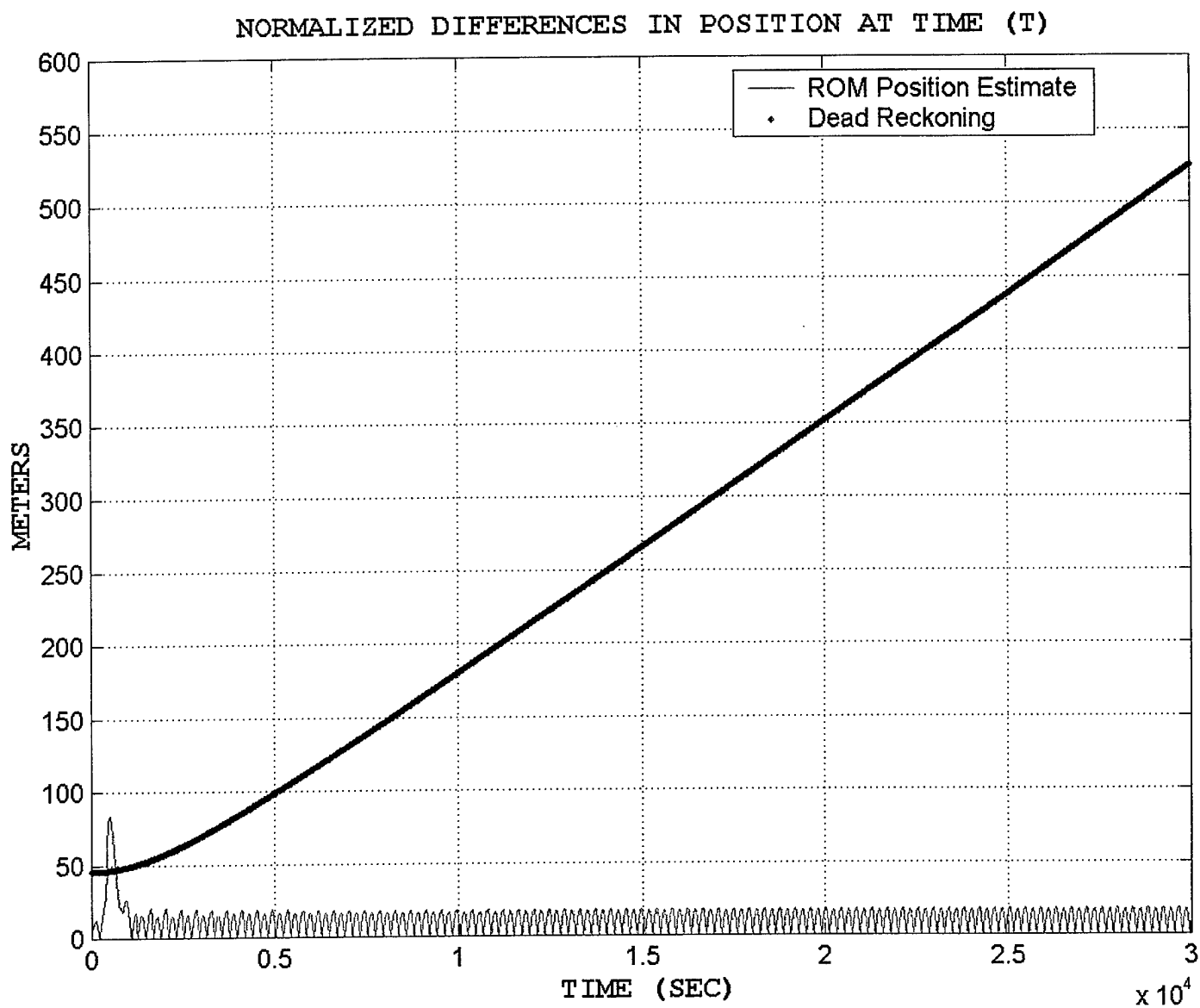


Figure 4.34 – Normalized differences between indicated and actual position for 100 meter “master” AUV path radius

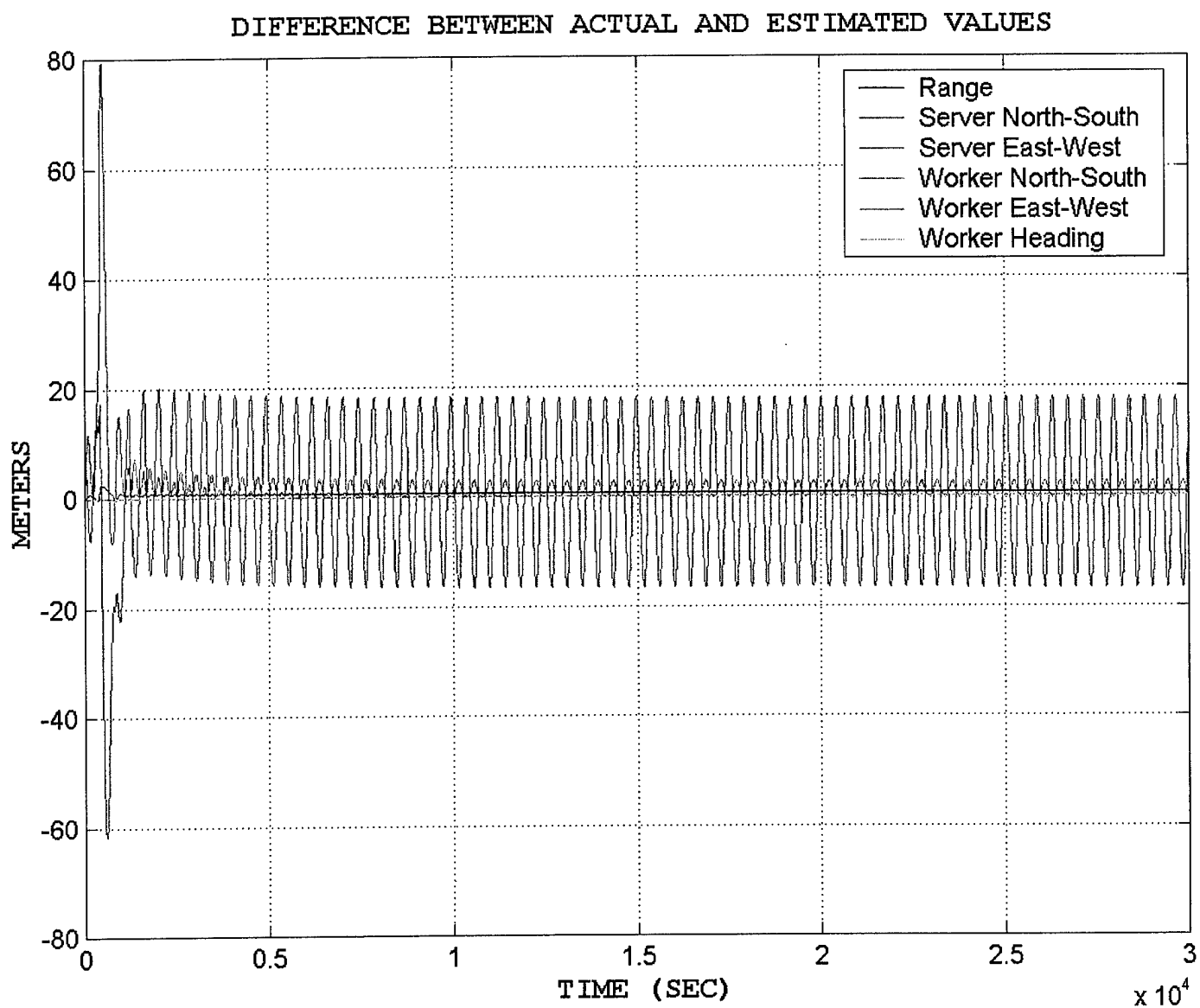


Figure 4.35 – Difference between actual and estimated position states for 100 meter “master” AUV path radius

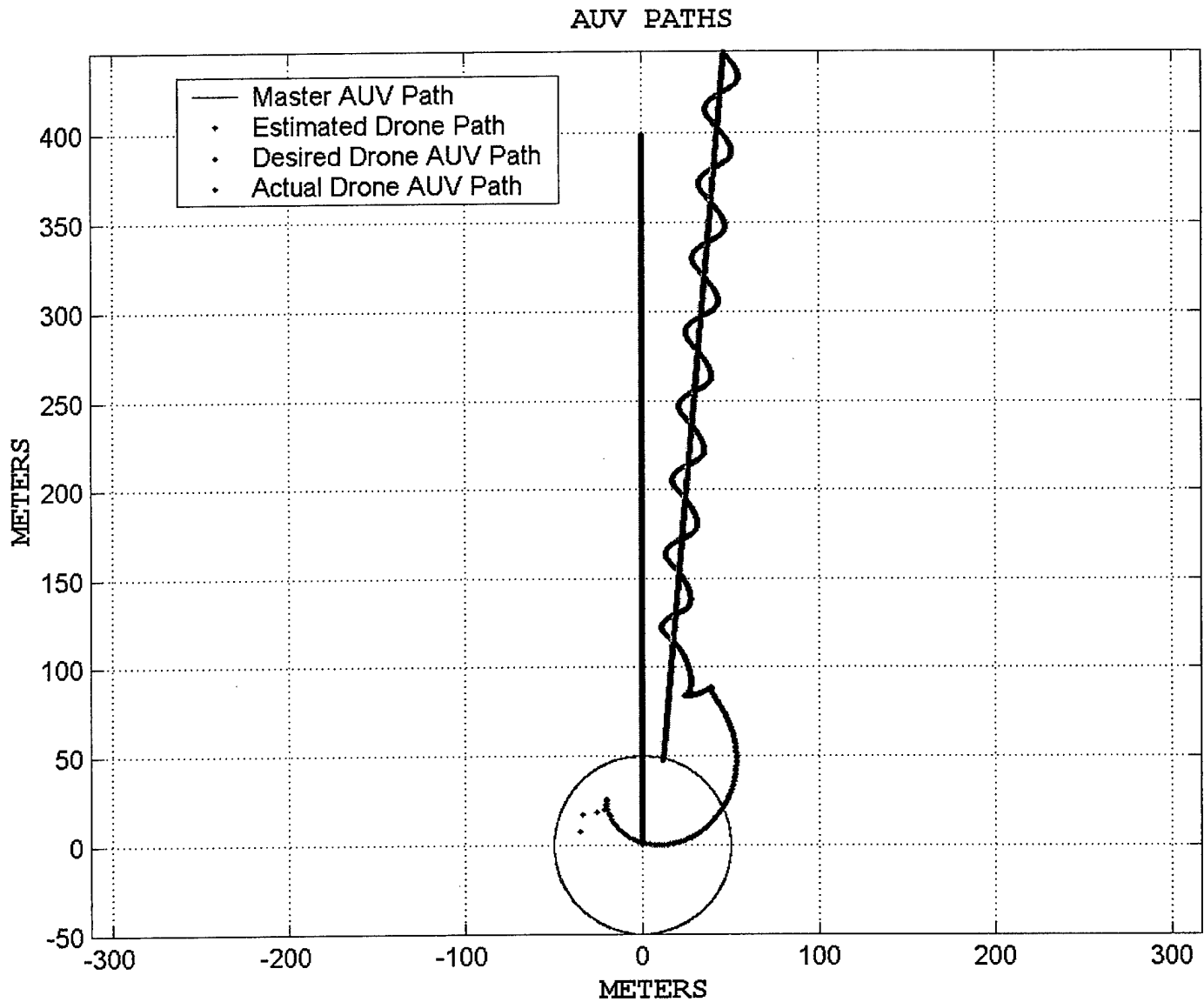


Figure 4.36 – AUV paths with no initial drone position data over 2000 seconds

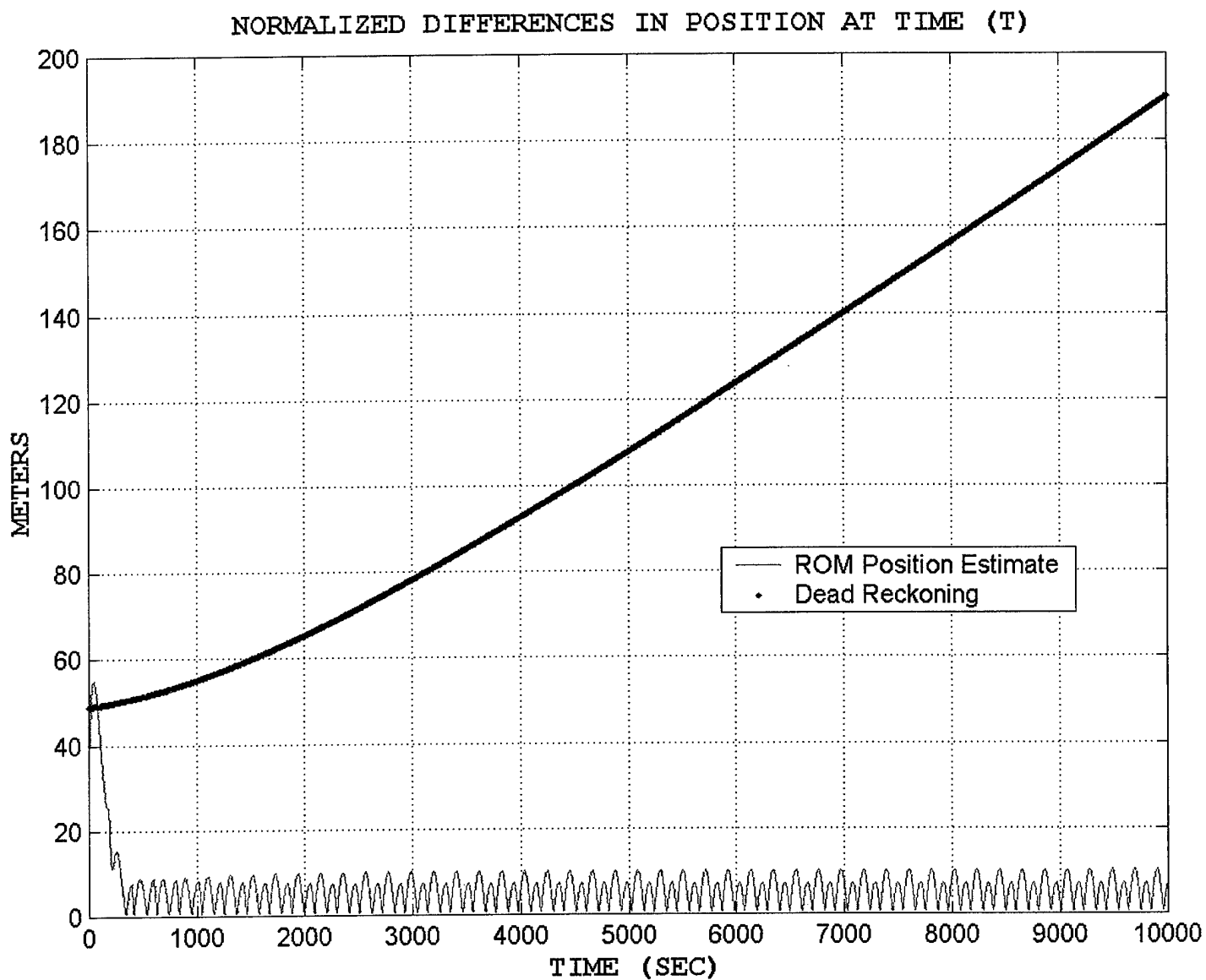


Figure 4.37 – Normalized differences between indicated and actual position for 50 meter “master” AUV path radius

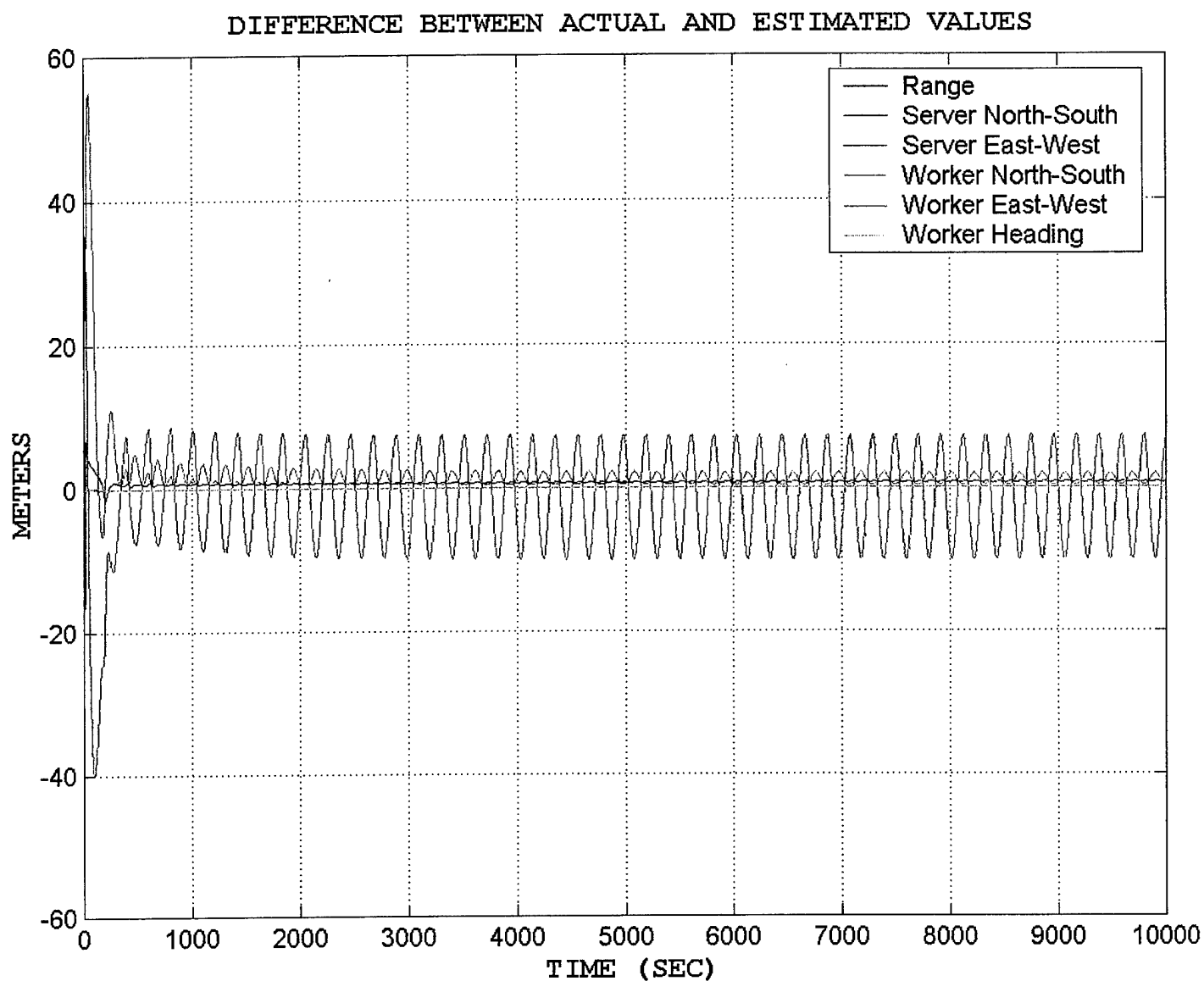


Figure 4.38 – Difference between actual and estimated position states for 50 meter “master” AUV path radius

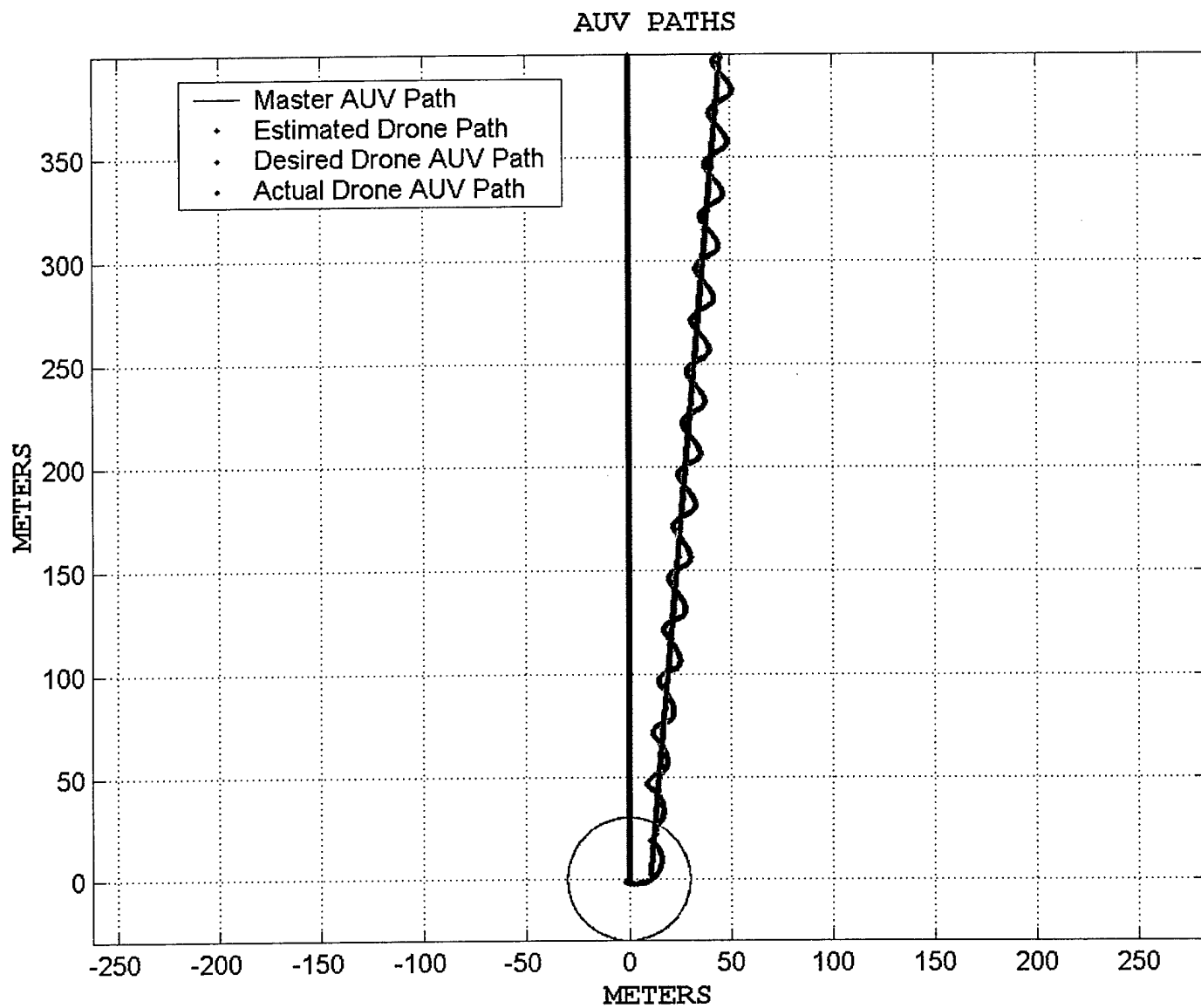


Figure 4.39 – AUV paths with no initial drone position data over 2000 seconds

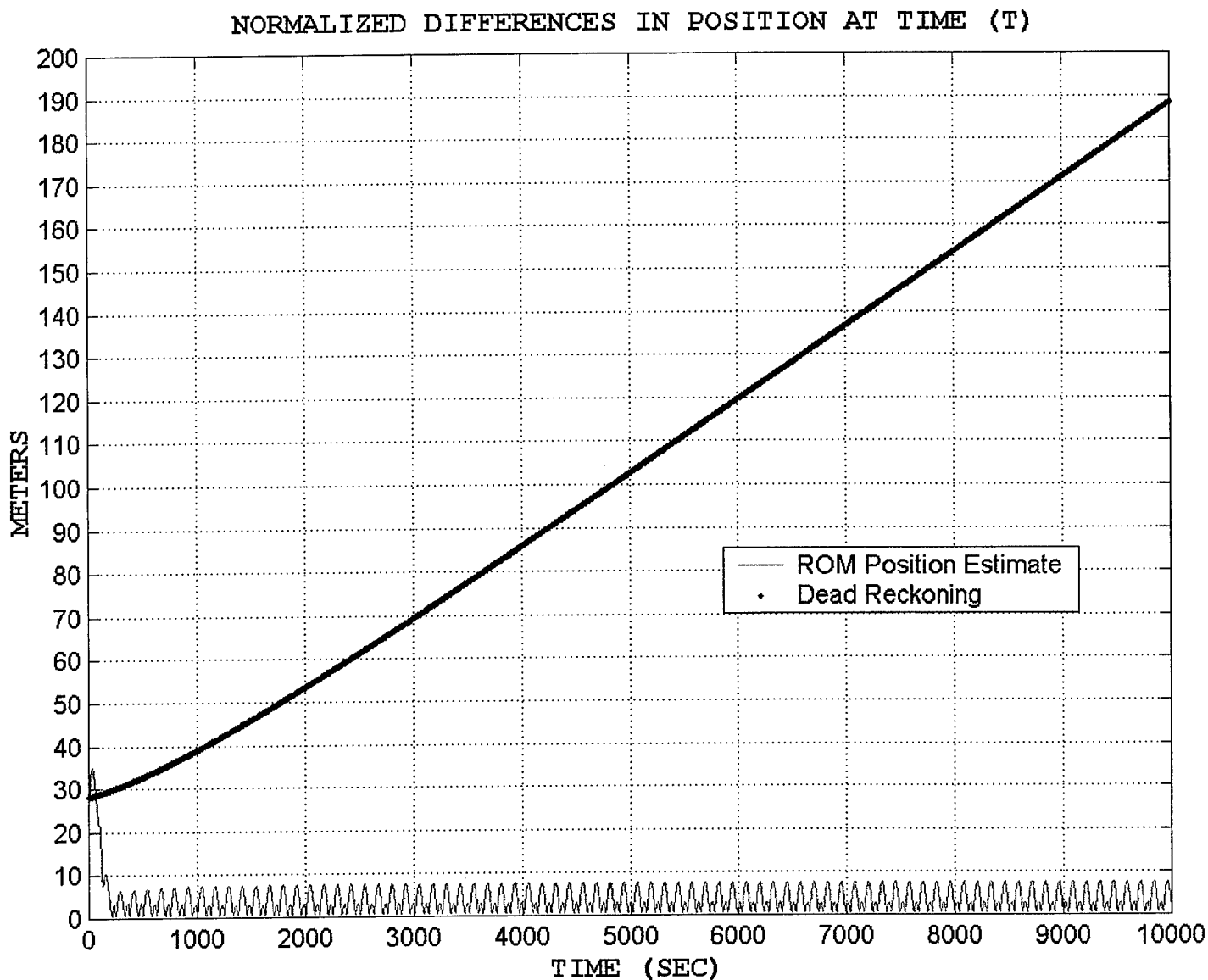


Figure 4.40 – Normalized differences between indicated and actual position for 30 meter “master” AUV path radius

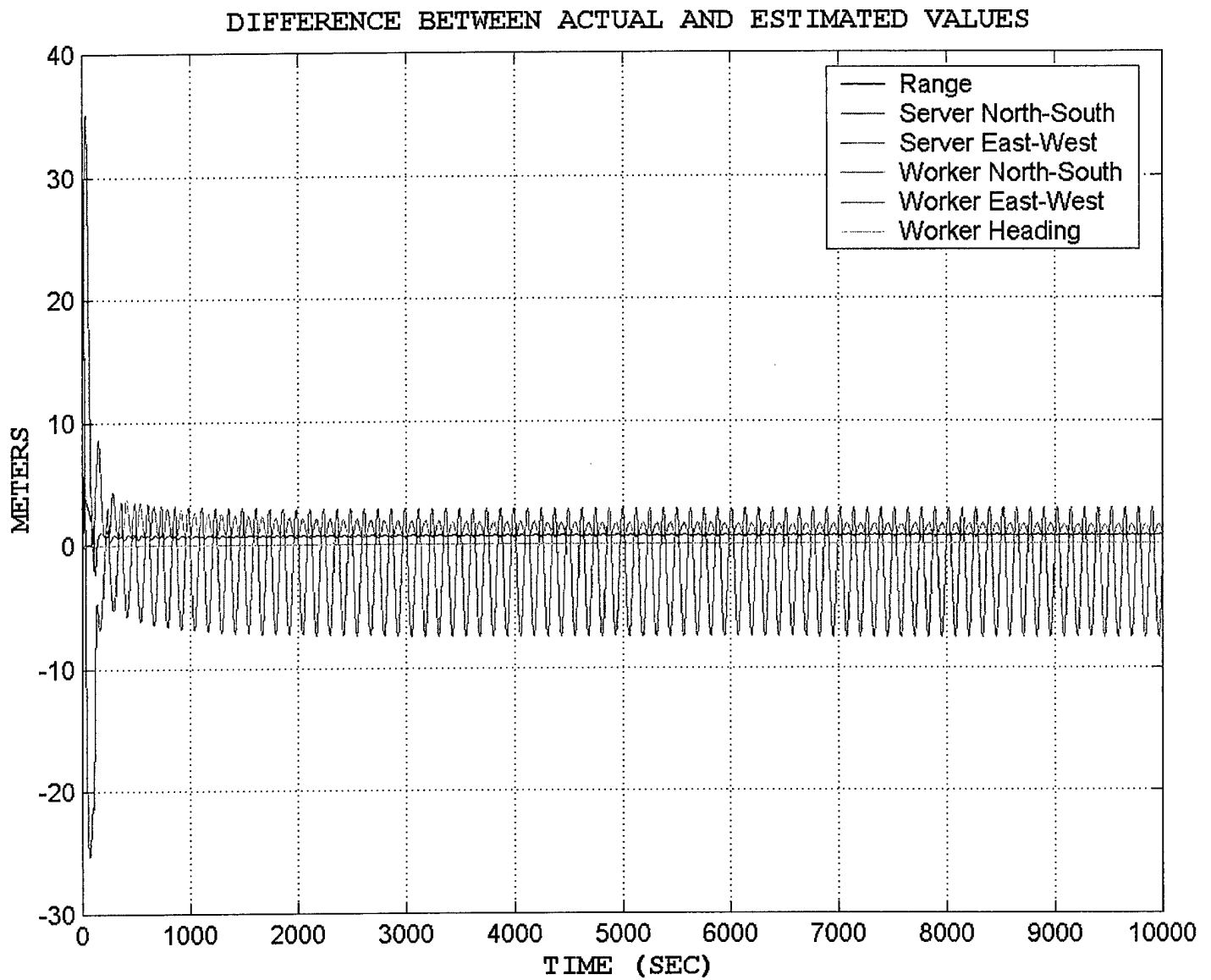


Figure 4.41 – Difference between actual and estimated position states for 30 meter “master” AUV path radius

AUV PATHS

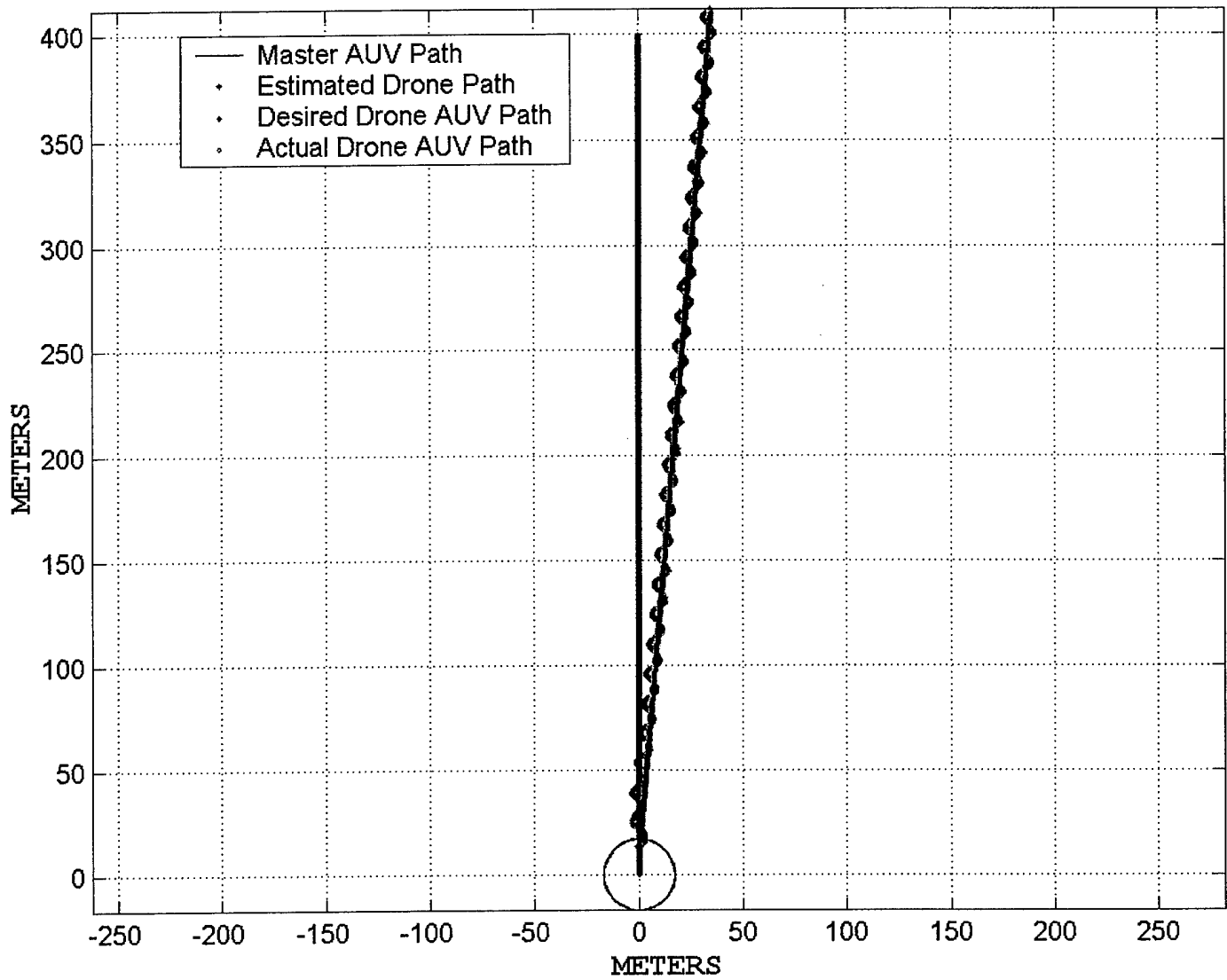


Figure 4.42 – AUV paths with no initial drone position data over 2000 seconds

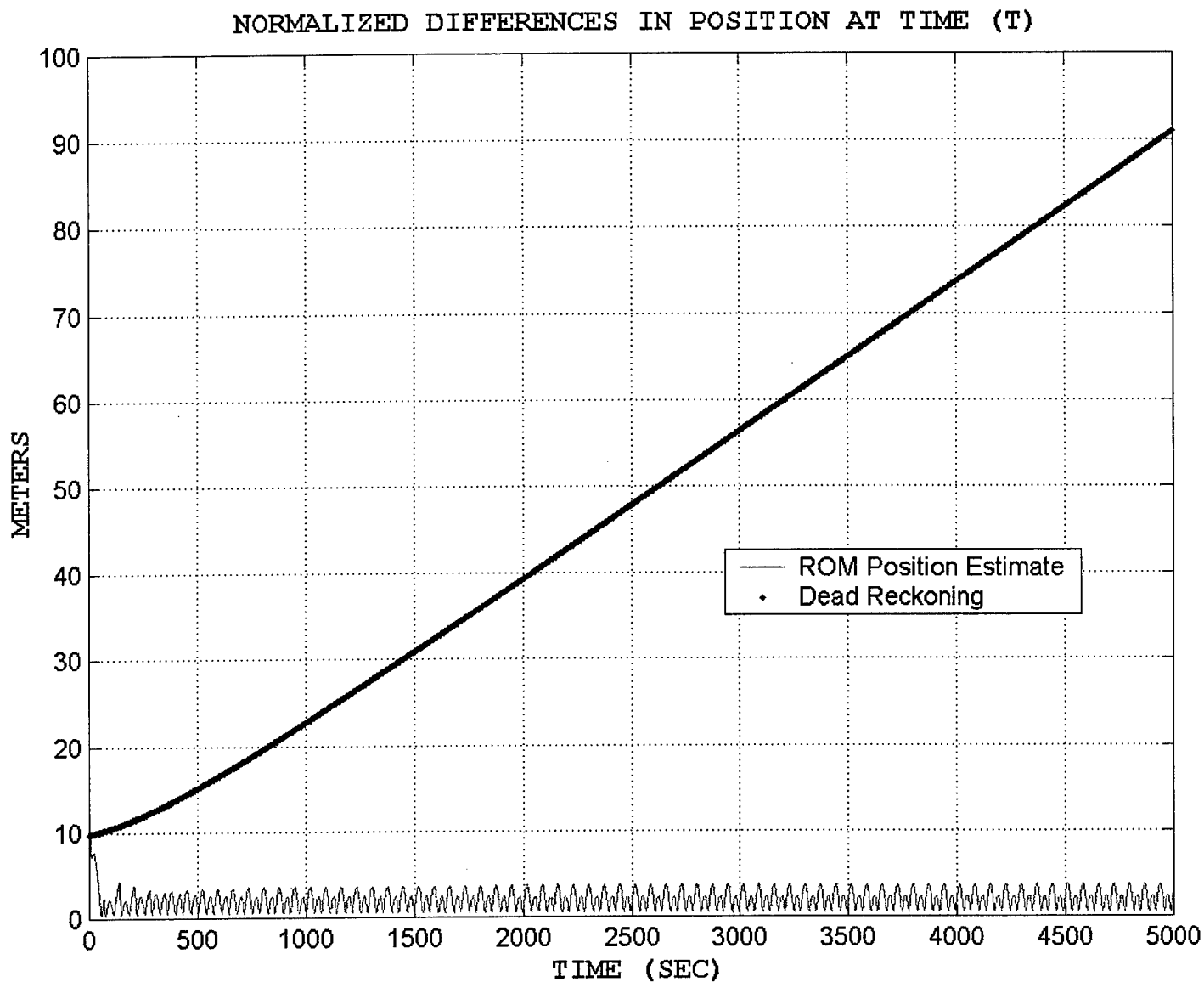


Figure 4.43 – Normalized differences between indicated and actual position for 17 meter “master” AUV path radius

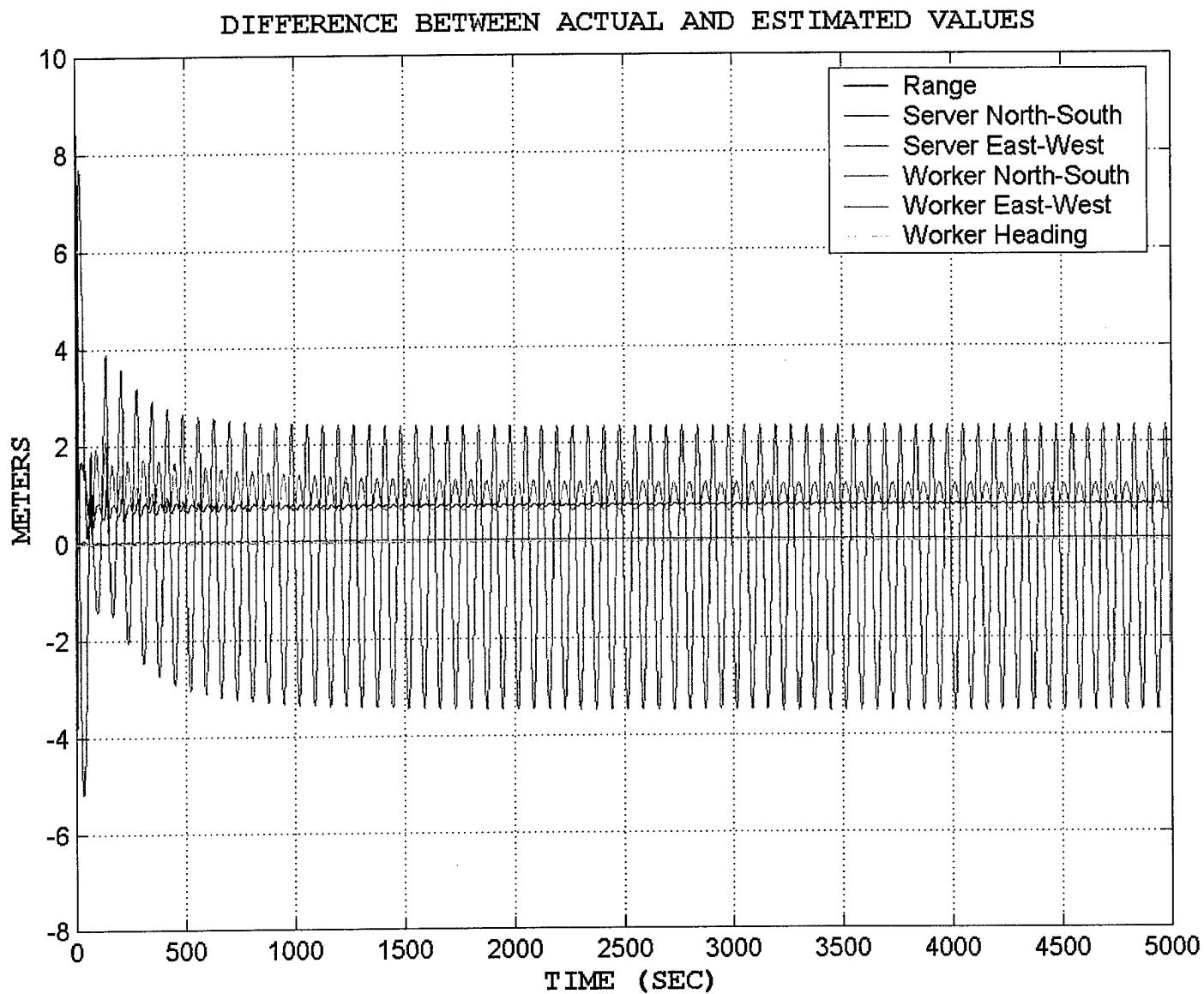


Figure 4.44 – Difference between actual and estimated position states for 17 meter
“master” AUV path radius

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

This work, through computer modeling and simulation, has shown that ROM position estimation will provide more accurate position estimates than dead reckoning. Results from this work also demonstrated that ROM position estimation produces observable results. From previous work, it was known that the server path must contain a non-zero acceleration motion to enable the EKF to find accurate, observable solutions (Song, 1999). However, the results also manifest that the accuracy of the solution is also dependent on the tracker vehicle's path. For the circular paths traveled by the master vehicle, larger non-zero accelerations translated to smaller circular paths higher bearing and range rates relative to the drone. As these rates increased, accuracy also increased.

B. RECOMMENDATIONS

The work and research completed in this thesis utilized simulations that were nearly ideal. Based on positive results gathered, further study should be made in both development of the EKF and the dynamic model.

Further optimization of the EKF to reduce the oscillatory errors would greatly improve estimated position accuracy. Also testing the EKF's accuracy when current and realistic noise values are modeled could prove very valuable in determining if this system could be used on an AUV.

Future research to enhance the dynamic model should include simulations with two or more drones acting independently or simulations with one drone making course

changes. Lastly, a vital issue requiring more research is finding alternate master vehicle paths that create observable solutions and that patrol operation areas effectively.

APPENDIX A. MATLAB CODE ASSOCIATED WITH KNOWN INITIAL POSITION SIMULATION

```

%Jason Alleyne
%initpos.m
%2 AUV Range only Measurement Simulation
%Known Initial Drone Position

clear
clc

%System dynamic model of worker measurement process for the range measurement

u=1.5; %server vehicle speed, u=1.5 m/s
Rs=100;
w=u/Rs; %Server vehicle path circular radius, and angular velocity
psi=0; %the worker vehicle ordered course
b=(pi/180)*5; %the worker vehicle compass bias, 5 degrees
q=[0.2;0;0]; %the worker speed is .2 m/s
Aw=[0,0,0;0,0,0;0,0,0];
x1=zeros(3,2000);x1a=zeros(3,2000); %initializing the actual (x1a) and dr (x1) position vectors
x1(:,1)=[0;0;0];x1a(:,1)=[0;0;0]; %initial posits/states at the origin for the simulations.
speed=zeros(1,2000);compass=zeros(1,2000); %initializing the speed and compass vectors
dt=1; %time step of one second
[p2,g2]=c2d(Aw,q,1); %continuous to discrete function for dr course

for i=1:(length(x1)-1);
    x1(:,i+1)=p2*x1(:,i)+g2;
    speed(i)=g2(1); %Initializes speed array for D.R.
    compass(i)=x1(3,i)+b; %Initializes compass array for D.R.
    f1(i)=speed(i)*cos(compass(i)); %The actual speed/heading due to compass bias
    f2(i)=speed(i)*sin(compass(i)); %The actual speed/heading due to compass bias
    q2=[f1(1,1);f2(1,1);0]; %Used as propagation q matrix
    [p3,g3]=c2d(Aw,q2,1); %C2D to convert continuous actual to discrete actual
    x1a(:,i+1)=p3*x1a(:,i)+g3; %Actual track of worker
    V(i)=x1(1,i)-x1a(1,i);X(i)=x1(2,i)-x1a(2,i);
    DIFF(i)=sqrt(V(i)^2+X(i)^2);
end

%circular motion of the server radius, Rs.
% x0 is the position vector [x,y] of the server
% Set up circular path for server. Example is radius of 100 meters at w=0.015 rad/sec.

%System dynamic model of server measurement process for the range measurement

As=[0,w;-w,0];q=[1;1];
x0=zeros(2,2000);x0(:,1)=[Rs;0];
[p,g]=c2d(As,q,1);
for i=1:(length(x0)-1);
    x0(:,i+1)=p*x0(:,i);
    R(i)=sqrt((x0(1,i)-x1(1,i))^2+(x0(2,i)-x1(2,i))^2); %Range between DR worker position and server
    %vehicle;
    Ra(i)=sqrt((x0(1,i)-x1a(1,i))^2+(x0(2,i)-x1a(2,i))^2); %actual Range between worker and server vehicles;
end

```

```
figure(2),clf,plot(x0(2,:),x0(1,:),b-',x1(2,:),x1(1,:),g'),grid
xlabel('\fontname{courier new}\fontsize{12}\bf METERS');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf AUV PATHS');
legend('Master AUV Path','Desired Drone AUV Path'),axis equal
```

```
figure(3),clf,plot(R,'g*'),grid
hold;plot(Ra,'b-');
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf RANGE (METERS)');
title('\fontname{courier new}\fontsize{12}\bf ACTUAL VS DEAD RECKONING RANGE');
legend('Dead Reckoning Range','Actual Range',0)
```

```
figure(4),clf,plot(x1(2,:),x1(1,:),g-',x1a(2,:),x1a(1,:),r*',x0(2,:),x0(1,:),b-'),grid,axis equal,
xlabel('\fontname{courier new}\fontsize{12}\bf METERS');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf AUV PATHS');
legend('Desired Drone AUV Path','Actual Drone AUV Path','Master AUV Path',0)
```

```
%build an Kalman filter for master vehicle
```

```
% Position Estimator based on data stream, R,
```

```
%
endSample=length(Ra);startSample=1;
y=[Ra;x0(1,1:length(Ra));x0(2,1:length(Ra));x1a(1,1:length(Ra));x1a(2,1:length(Ra));x1a(3,1:length(Ra))];
x=zeros(5,endSample);err=zeros(6,endSample);
s=startSample;
x(:,s)=[x0(1,1),x0(2,1),0,0,0];
```

```
%MANEUVERING AND CURRENT TIME CONSTANTS
```

```
tau1=0;
tau2=0;
```

```
A=[As,zeros(2,3);zeros(3,2),Aw];
```

```
C=zeros(6,5); %Initializes the C matrix
C(2,1)=1;C(3,2)=1;C(4,3)=1;C(5,4)=1;C(6,5)=1; %Matches measurement states (y) to output states (x)
```

```
Ra(1)=sqrt((x0(1,1)-x1a(1,1))^2+(x0(2,1)-x1a(2,1))^2);
```

```
% C matrix local is dg/dx
C(1,3)=-(x0(1,1)-x1a(1,1))/Ra(1);
C(1,4)=-(x0(2,1)-x1a(2,1))/Ra(1);
C(1,1)=(x0(1,1)-x1a(1,1))/Ra(1);
C(1,2)=(x0(2,1)-x1a(2,1))/Ra(1);
```

```
%Initial B matrix
```

```
q1=0.01; %variance on X, m^2
q2=0.01; %variance on Y, m^2
q3=0.1; %variance on x1a
q4=.1; %variance on y1a, rad/s)^2
q5=0.0001; %slow bias convergence
```

```

B=[q1;q2;q3;q4;q5]/100;

Q=diag(B); %system noise

nu1=0.01;nu2=.001;nu3=.001;nu4=2000;nu5=2000;nu6=.02;

gnu=[nu1;nu2;nu3;nu4;nu5;nu6];

R=diag(gnu)*1; %measurement noise
Gram=zeros(5,5); %initialization of observability grammian matrix

% measured data at old time, new_before means model predicted value

p_old_after=eye(5);
delx_old_after=zeros(5,1);

g=ones(6,1);psave=zeros(5,length(R));

dt=1;phi=expm(A*dt); t=0:1:endSample-1;

for i=startSample:(endSample-1);

%compute linearized PHI matrix using updated A

%reset initial state

x_old_after=x(:,i);

% nonlinear state propagation

[x_new_before]=phi*x_old_after;

%error covariance propagation

p_new_before=phi*p_old_after*phi' + Q; %new gain calculation using linearized new
%C matrix and current state error
%covariances.

%formulate the innovation using nonlinear output propagation
%as compared to new sampled data from measurements

yhat=output3(x_new_before);
err(:,i+1)=(y(:,i+1) - yhat);

G=p_new_before*C*inv(C*p_new_before*C' + R); % Kalman Gain
p_old_after=[eye(5) - G*C]*p_new_before;
psave(:,i+1)=diag(p_old_after);

cpc=inv(C*p_old_after*C'+R);
rel(i+1)=err(:,i+1)*cpc*err(:,i+1);

x_new_after=x_new_before + G*err(:,i+1);

```

```

%carry new state into next update

x(:,i+1)=x_new_after;

C=zeros(6,5);
C(2,1)=1;C(3,2)=1;C(4,3)=1;C(5,4)=1;C(6,5)=1;    %Current settings make all target states observable to
                                                    %filter.

% Cmatrix local is dg/dx

C(1,3)=-(x(1,i)-x(3,i))/y(1,i);
C(1,4)=-(x(2,i)-x(4,i))/y(1,i);
C(1,1)=-C(1,3);
C(1,2)=-C(1,4);
O(i)=rank([C',phi'*C]);
U(i)=x(4,i)-y(5,i);T(i)=x(3,i)-y(4,i);
diff(i)=sqrt(U(i)^2+T(i)^2);
Gram=Gram+phi'*C*inv(R)*C*phi;
Gramrank(i)=rank(Gram);
end;

figure(5),clf,plot(O,'b*'),grid,zoom %Local Observability indicates # of observable states at a point in
                                     %time.
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf OBSERVABLE STATES');
title('\fontname{courier new}\fontsize{12}\bf LOCAL SYSTEM OBSERVABILITY')

figure(6),clf,plot(Gramrank,'b*'),grid,zoom %Observability Grammian Rank indicates # of observable
                                             %states over period of the simulation
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf OBSERVABLE STATES');
title('\fontname{courier new}\fontsize{12}\bf TOTAL SYSTEM OBSERVABILITY')

figure(7),clf,plot(t,rel,'b*'),grid,zoom

figure(8),clf,plot(t,x(4,:), 'b*',t,x1a(2,1:1999), 'm.'),grid %East-West position comparison
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf ACTUAL vs. ESTIMATED EAST-WEST POSITION');
legend('Estimated position','Actual position',0)

figure(9),clf,plot(t,x(3,:), 'b*',t,x1a(1,1:1999), 'm.'),grid %North-south position comparison
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf ACTUAL vs. ESTIMATED NORTH-SOUTH POSITION');
legend('Estimated position','Actual position',0)

figure(10),clf,plot(x(2,:),x(1,:)),grid,hold,
plot(x(4,:),x(3,:), 'r.'),plot(x1(2,:),x1(1,:), 'm.'),plot(x1a(2,1:1999),x1a(1,1:1999), 'c.'),zoom,hold off,axis
equal
xlabel('\fontname{courier new}\fontsize{12}\bf METERS');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf AUV PATHS');
legend('Master AUV Path','Estimated Drone Path','Desired Drone AUV Path','Actual Drone AUV Path',0)

```

```

figure(11),clf,plot(err'),grid,
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf DIFFERENCE BETWEEN ACTUAL AND ESTIMATED
VALUES');
legend('Range','Server North-South','Server East-West','Worker North-South','Worker East-West','Worker
Heading',0)

```

```

figure(12),clf,plot(diff,'r-'),grid,hold,plot(DIFF,'b.'),hold off
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf NORMALIZED DIFFERENCES IN POSITION AT TIME
(T)');
legend('ROM Position Estimate','Dead Reckoning',0)

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. *MATLAB* CODE ASSOCIATED WITH UNKNOWN INITIAL POSITION SIMULATION

```

%Jason Alleyne
%no-initpos.m
%2 AUV Range only Measurement Simulation
%Unknown Initial Drone Position

clear
clc

%System dynamic model of worker measurement process for the range measurement

u=1.5; %server vehicle speed, u=1.5 m/s
Rs=100;w=u/Rs; %Server vehicle path circular radius, and angular velocity
psi=0; %the worker vehicle ordered course
b=(pi/180)*5; %the worker vehicle compass bias, 5 degrees
q=[0.2;0;0]; %the worker speed is .2 m/s
Aw=[0,0,0;0,0,0;0,0,0];
x1=zeros(3,2000);x1a=zeros(3,2000); %initializing the actual (x1a) and dr (x1) position vectors
x1(:,1)=[0;0;0];l=randn([2,1])*Rs;x1a(:,1)=[l;0]; %The actual position is unknown, and is a random
initial value.
speed=zeros(1,2000);compass=zeros(1,2000); %initializing the speed and compass vectors
dt=1; %time step of one second
[p2,g2]=c2d(Aw,q,1); %continuous to discrete function for dr course

for i=1:(length(x1)-1);
    x1(:,i+1)=p2*x1(:,i)+g2;
    speed(i)=g2(1); %Initializes speed array for D.R.
    compass(i)=x1(3,i)+b; %Initializes compass array for D.R.
    f1(i)=speed(i)*cos(compass(i)); %is the actual speed/heading due to compass bias
    f2(i)=speed(i)*sin(compass(i)); %is the actual speed/heading due to compass bias
    q2=[f1(1,1);f2(1,1);0]; %used as propagation q matrix
    [p3,g3]=c2d(Aw,q2,1); %C2D to convert continuous actual to discrete actual
    x1a(:,i+1)=p3*x1a(:,i)+g3; %actual track of worker
    V(i)=x1(1,i)-x1a(1,i);X(i)=x1(2,i)-x1a(2,i);
    DIFF(i)=sqrt(V(i)^2+X(i)^2);
end

%circular motion of the server radius, Rs.
% x0 is the position vector [x,y] of the server
% Set up circular path for server. Example is radius of 100 metres at w=0.015 rad/sec.

%System dynamic model of server measurement process for the range measurement

As=[0,w;-w,0];q=[1;1];
x0=zeros(2,2000);x0(:,1)=[Rs;0];
[p,g]=c2d(As,q,1);
for i=1:(length(x0)-1);
    x0(:,i+1)=p*x0(:,i);
    R(i)=sqrt((x0(1,i)-x1(1,i))^2+(x0(2,i)-x1(2,i))^2); %Range between estimated worker position and
    %server vehicle;
    Ra(i)=sqrt((x0(1,i)-x1a(1,i))^2+(x0(2,i)-x1a(2,i))^2); %actual Range between worker and server vehicles
end

```

```
figure(1),clf,plot(x0(2,:),x0(1,:), 'b-',x1(2,:),x1(1,:), 'g-'),grid
xlabel('\fontname{courier new}\fontsize{12}\bf METERS');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf AUV PATHS');
legend('Master AUV Path','Desired Drone AUV Path'),axis equal
```

```
figure(2),clf,plot(R,'g*'),grid
hold;plot(Ra,'b-');
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf RANGE (METERS)');
title('\fontname{courier new}\fontsize{12}\bf ACTUAL VS DEAD RECKONING RANGE');
legend('Dead Reckoning Range','Actual Range',0)
```

```
figure(3),clf,plot(x1(2,:),x1(1,:), 'g-',x1a(2,:),x1a(1,:), 'r*',x0(2,:),x0(1,:), 'b^'),grid,axis equal,
xlabel('\fontname{courier new}\fontsize{12}\bf METERS');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf AUV PATHS');
legend('Desired Drone AUV Path','Actual Drone AUV Path','Master AUV Path',0)
```

%build a Kalman filter for server vehicle

```
% Position Estimator based on data stream, R,
% _____
endSample=length(Ra);startSample=1;
y=[Ra;x0(1,1:length(Ra));x0(2,1:length(Ra));x1a(1,1:length(Ra));x1a(2,1:length(Ra));x1a(3,1:length(Ra))];
x=zeros(5,endSample);err=zeros(6,endSample);
s=startSample;
x(:,s)=[x0(1,1),x0(2,1),0,0,0];
```

%MANEUVERING AND CURRENT TIME CONSTANTS

```
tau1=0;
tau2=0;
```

```
A=[As,zeros(2,3);zeros(3,2),Aw];
```

```
C=zeros(6,5);
C(2,1)=1;C(3,2)=1;C(4,3)=0;C(5,4)=0;C(6,5)=0;
```

%Initializes the C matrix
%Matches measurement states (y) to output states (x)
%with drone states unobservable

```
Ra(1)=sqrt((x0(1,1)-x1a(1,1))^2+(x0(2,1)-x1a(2,1))^2);
```

```
% Cmatrix local is dg/dx
C(1,3)=-(x0(1,1)-x1a(1,1))/Ra(1);
C(1,4)=-(x0(2,1)-x1a(2,1))/Ra(1);
C(1,1)=(x0(1,1)-x1a(1,1))/Ra(1);
C(1,2)=(x0(2,1)-x1a(2,1))/Ra(1);
```

%Initial B matrix

```
q1=0.01;
q2=0.01;
q3=0.1;
q4=.1;
q5=0.0001;
```

%variance on X, m^2
%variance on Y, m^2
%variance on x1a
%variance on y1a, rad/s)^2
%slow bias convergence


```

B=[q1;q2;q3;q4;q5]/100;

Q=diag(B);

%system noise

nu1=0.01;nu2=.001;nu3=.001;nu4=2000;nu5=2000;nu6=.02;

gnu=[nu1;nu2;nu3;nu4;nu5;nu6];

R=diag(gnu)*1; %measurement noise
Gram=zeros(5,5); %initialization of observability grammian matrix

% measured data at old time, new_before means model predicted value

p_old_after=eye(5);
delx_old_after=zeros(5,1);

g=ones(6,1);psave=zeros(5,length(R));

dt=1;phi=expm(A*dt); t=0:1:endSample-1;

for i=startSample:(endSample-1);

%compute linearized PHI matrix using updated A

%reset initial state

x_old_after=x(:,i);

% nonlinear state propagation

[x_new_before]=phi*x_old_after;

%error covariance propagation

p_new_before=phi*p_old_after*phi' + Q; %new gain calculation using
%linearized new C matrix and
%current state error covariances.

%formulate the innovation using nonlinear output propagation
%as compared to new sampled data from measurements

yhat=output3(x_new_before);
err(:,i+1)=(y(:,i+1) - yhat);

G=p_new_before*C*inv(C*p_new_before*C' + R); % Kalman Gain
p_old_after=[eye(5) - G*C]*p_new_before;
psave(:,i+1)=diag(p_old_after);

cpc=inv(C*p_old_after*C'+R);
rel(i+1)=err(:,i+1)*cpc*err(:,i+1);

x_new_after=x_new_before + G*err(:,i+1);

```

```

%carry new state into next update

x(:,i+1)=x_new_after;

C=zeros(6,5);
C(2,1)=1;C(3,2)=1;C(4,3)=0;C(5,4)=0;C(6,5)=1;

% Cmatrix local is dg/dx

C(1,3)=-(x(1,i)-x(3,i))/y(1,i);
C(1,4)=-(x(2,i)-x(4,i))/y(1,i);
C(1,1)=-C(1,3);
C(1,2)=-C(1,4);
O(i)=rank([C',phi*C']);
U(i)=x(4,i)-y(5,i);T(i)=x(3,i)-y(4,i);
diff(i)=sqrt(U(i)^2+T(i)^2);
Gram=Gram+phi*C'*inv(R)*C*phi;
Gramrank(i)=rank(Gram);
end;

figure(4),clf,plot(O,'b*'),grid %Local Observability indicates # of observable states at a point in time.
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf OBSERVABLE STATES');
title('\fontname{courier new}\fontsize{12}\bf LOCAL SYSTEM OBSERVABILITY')

figure(5),clf,plot(Gramrank,'b*'),grid,axis([0 length(x0) 4 6]) %Observability Grammian Rank indicates
                                                                    %of observable states over period of the
                                                                    %simulation
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf OBSERVABLE STATES');
title('\fontname{courier new}\fontsize{12}\bf TOTAL SYSTEM OBSERVABILITY')

figure(6),clf,plot(t,rel,'b*'),grid,zoom

figure(7),clf,plot(t,x(4,:), 'b*',t,x1a(2,1:1999),'m.'),grid %East-West position comparison
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf ACTUAL vs. ESTIMATED EAST-WEST POSITION');
legend('Estimated position','Actual position',0)

figure(8),clf,plot(t,x(3,:), 'b*',t,x1a(1,1:1999),'m.'),grid %North-south position comparison
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf ACTUAL vs. ESTIMATED NORTH-SOUTH POSITION');
legend('Estimated position','Actual position',0)

figure(9),clf,plot(x(2,:),x(1,:)),grid,hold,
plot(x(4,:),x(3,:), 'r.'),plot(x1(2,:),x1(1,:), 'm.'),plot(x1a(2,1:1999),x1a(1,1:1999),'c. ');hold off,axis
equal,zoom
xlabel('\fontname{courier new}\fontsize{12}\bf METERS');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf AUV PATHS');
legend('Master AUV Path','Estimated Drone Path','Desired Drone AUV Path','Actual Drone AUV Path',0)

```

```

figure(10),clf,plot(err'),grid,
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf DIFFERENCE BETWEEN ACTUAL AND ESTIMATED
VALUES');
legend('Range','Server North-South','Server East-West','Worker North-South','Worker East-West','Worker
Heading',0)

```

```

figure(11),clf,plot(diff,'r-'),grid,hold,plot(DIFF,'b. '),hold off
xlabel('\fontname{courier new}\fontsize{12}\bf TIME (SEC)');
ylabel('\fontname{courier new}\fontsize{12}\bf METERS');
title('\fontname{courier new}\fontsize{12}\bf NORMALIZED DIFFERENCES IN POSITION AT TIME
(T)');
legend('ROM Position Estimate','Dead Reckoning',0)

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. *MATLAB* CODE ASSOCIATED WITH EXTENDED KALMAN FILTER MEASUREMENT DATA FUSION

```
%Jason Alleyne
%Output3.m
%Function required for ROM EKF used in initpos.m and no-initpos.m

function [yhat]=output3(xold);

x0=xold(2);
y0=xold(1);
x1=xold(4);
y1=xold(3);
psi1=xold(5);    %ensure the states are in the same order as in main program

Rhat=sqrt((x0-x1)^2+(y0-y1)^2);
x0hat=x0;
y0hat=y0;

yhat=[Rhat;y0;x0;y1;x1;psi1];
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- Bishop, G., Welch, G., "An Introduction to the Kalman Filter," <http://ftp.cs.brown.edu/stc/education/course95-96/Kalman-Filters/>, November, 1995.
- Healey, A.J., An, E.A., Marco, D.B., "On Line Compensation of Heading Sensor Bias for Low Cost AUV's," Proceedings of the IEEE Workshop on Autonomous Underwater Vehicles, AUV98, IEEE Catalog, Number 98CH36290, ISBN # 0-7803-5190-8, August 20-21, 1998, Cambridge, Mass. Pp 35-42.
- Little, J.N., *Control Toolbox*, The Math Works, Inc. April 1995.
- Marco, D.B., Healey, A.J., "Currents Developments in Underwater Vehicle Control and Navigation: The NPS ARIES AUV," Proceedings of IEEE Workshop Oceans 2000, September 11-14, 2000, Providence, RI.
- Song, T.L., "Observability of Target Tracking with Range-Only Measurements," IEEE Journal of Oceanic Engineering, Vol. 24, No. 3, July 1999, pp.383-387.
- Stinespring, B.M., "The Experimental Evaluation of a DGPS Based Navigation Suite in the ARIES AUV," MSME Thesis, Naval Postgraduate School, Monterey, CA. June 2000.
- Zhengyou Zhang, "Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting," <http://www-sop.inria.fr/robotvis/personnel/zzhang/Publis/Tutorial-Estim/node1.html>, February, 1996.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
8725 John J. Kingman Road, Suite 0944
Ft. Belvoir, VA 22060-6218
2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Road
Monterey, CA 93943-5101
3. Professor Anthony J. Healey, Code ME/Hy..... 2
Mechanical Engineering Department
Naval Postgraduate School
Monterey, California 93943-5101
4. Mechanical Engineering Department Chairman, Code ME..... 1
Naval Postgraduate School
Monterey, California 93943-5101
5. Engineering & Technology Curricular Office, Code 34 1
Naval Postgraduate School
Monterey, California 93943-5101
6. Dr. Tom Curtin..... 1
Office of Naval Research
800 N. Quincy St
Arlington, VA, 22217-5660
7. Dr. Edgar An 1
Florida Atlantic University
Sea Tech Campus
Institute of Ocean and Systems Engineering
101 North Dania Beach Rd.
Dania Beach, FL, 33004
8. Dr. Chris Von Alt 1
Woods Hole Oceanographic Institution
Deep Submergence Laboratory
Woods Hole, Ma, 02354

9. Dr. Antonio Pascoal 1
Institute Superior Tecnico
Av. Rovisco Pais,
1096 Lisboa Codex, PORTUGAL
10. LT Jason C. Alleyne 1
14 Sylvia Rd.
North Reading, MA 01864
11. Dr. Tom Swean 1
Office of Naval Research
800 N. Quincy St.
Arlington, VA 22217-5660